

VŠB – Technická univerzita Ostrava
Fakulta elektrotechniky a informatiky
Katedra elektroniky

ŘÍDICÍ JEDNOTKA PULZNÍHO MĚNIČE
S MIKROPOČÍTAČEM
DC-DC CONVERTER MICROCOMPUTER
CONTROL UNIT

2017

Martin Kastl

VŠB - Technická univerzita Ostrava
Fakulta elektrotechniky a informatiky
Katedra elektroniky

Zadání bakalářské práce

Student:

Martin Kastl

Studijní program:

B2649 Elektrotechnika

Studijní obor:

2612R003 Aplikovaná elektronika

Téma:

Řídicí jednotka pulzního měniče s mikropočítačem
DC-DC Converter Microcomputer Control Unit

Jazyk vypracování:

čeština

Zásady pro vypracování:

1. Proved'te popis výukového systému s mikropočítačem řady 8051 s ohledem na možnost připojení výkonové jednotky pulzního měniče.
2. Proved'te rozbor metod řízení výkonové jednotky s ohledem na typ zátěže.
3. Sestavte software pro ověření metod řízení jednotky s jednotlivými typy zátěže.
4. Proved'te měření průběhů důležitých veličin dle pokynů vedoucího závěrečné práce.

Seznam doporučené odborné literatury:


Dle pokynů vedoucího závěrečné práce.

Formální náležitosti a rozsah bakalářské práce stanoví pokyny pro vypracování zveřejněné na webových stránkách fakulty.

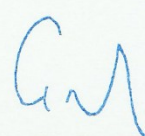
Vedoucí bakalářské práce: **doc. Ing. Petr Palacký, Ph.D.**

Datum zadání: 01.09.2016

Datum odevzdání: 28.04.2017


doc. Ing. Petr Palacký, Ph.D.
vedoucí katedry




prof. RNDr. Václav Snášel, CSc.
děkan fakulty

Prohlášení

*Prohlašuji, že jsem tuto bakalářskou práci vypracoval samostatně.
Uvedl jsem všechny literární prameny a publikace, ze kterých jsem čerpal.*

V Ostravě dne 28. 4. 2017



Poděkování

Rád bych poděkoval doc. Ing. Petru Palackému, Ph.D. za cenné rady, věcné připomínky a vstřícnost při konzultacích a vypracování bakalářské práce.

Abstrakt

Tato bakalářská práce řeší realizaci softwaru pro mikrokontrolér AT89S52, který je následně připojen k výkonové jednotce pulzního měniče. Na výstup pulzního měniče se poté připojí odporová a motorická zátěž. Zdrojový kód byl celý vytvořen v assembleru pomocí programu Keil. Algoritmus pro řízení pulzního měniče je bipolární i unipolární. Podařilo se docílit velmi velké přesnosti požadované frekvence a střidy. Odchylka není větší než 1%.

První kapitola popisuje výukový systém, na kterém se nachází mikrokontrolér AT89S52, co všechno obsahuje a základní parametry. Další kapitola charakterizuje výkonovou jednotku, princip pulzního měniče, metody řízení a jejich rozdíly. V třetí části je popsán software, zobrazeny diagramy programu a také ovládání. Následující a poslední část zobrazuje naměřené výsledky. V této kapitole se ověřuje funkčnost softwaru a popisují důležité veličiny při různých zátěžích.

Klíčová slova

Mikroprocesor, mikrokontrolér AT89S52, pulzní měnič, bipolární metoda, unipolární metoda, pulzně šířková modulace, PŠM

Abstract

This bachelor thesis solves the implementation of software for microcontroller AT89S52, which is then connected to the DC-DC converter. On the output of DC-DC converter is connected to resistive and motoric loads. Source code was created in the assembler using the program Keil. Algorithm for controlling DC-DC converter is bipolar and unipolar. I was able to achieve very high accuracy desired frequency and duty cycle. Deviation is not greater than 1%.

The first chapter describes the educational system which includes microcontroller AT89S52, his contain and basic parameters. Next chapter characterizes the performance unit, principle of DC-DC unit, methods of control and their differences. In the third part is described software, displayed diagrams and also control. Next and last part shows the measured results. In this chapter I verified the functionality of the software and describes the important value at different loads.

Keywords

Microprocessor, microcontroller AT89S52, DC-DC convertor, bipolar method, unipolar method, pulse width modulation, PWM

Seznam použitých symbolů a značek

Použité zkratky a pojmy

ACC	registr A - akumulátor
A/D	analogově digitální převodník
ALE	Adress Latch Enable – používá se pro multiplexování
AND	logický součin
B	registr B
BCD	dekodér pro sedmissegmentový displej
BI	bipolární metoda
CLK	hodinový signál
C/ \bar{T}	slouží k určení režimu čítač nebo časovač
D/A	digitálně analogový převodník
DPL, DPH	dolní a horní registr instrukčního ukazatele
EA	External Access – výběr čtení programu z vnitřní nebo vnější paměti
ES	Enable All - povolení všech přerušení
ET0, ET1	povolení přerušení od vysílání a příjmu sériového kanálu
EX0, EX1	povolení přerušení čítače/časovače 0 (1)
FCE	povolení vnějšího přerušení INT0 (INT1)
GATE	název tlačítka pro změnu funkce
HA, HB, DA, DB	k určení vnějšího nebo vnitřního zdroje pro spuštění a vypnutí čítače/časovače
IE	řídící signály tranzistorů
IE0, IE1	Interrupt Enabled – registr pro povolení přerušení
INT, INT0, INT1	povolení vnějšího přerušení
IP	vnější vstup mikrokontroléru pro přerušení
IT, IT0, IT1	Interrupt Priority – registr pro určení priority přerušení
M0, M1	nastavení přerušení na sestupnou hranu signálu nebo log. 0
N/P	slouží k určení módu čítače/časovače
OE(G)	název tlačítka pro změnu zobrazené hodnoty z paměti nebo potenciometru
OUT+, OUT-	pin na posilovači sběrnice pro aktivaci LED diod
P0, P1, P2, P3	konektory pro napájení pulzního měniče
PCON	brány mikroprocesoru
PM	Power Management Register – registr pro správu napájení
PS	pulzní měnič
PSEN	priorita sériového kanálu
PSW	Program Store Enable – signál při čtení programu z externí paměti
PT0, PT1	Processor Status Word – registr pro fungování mikroprocesoru
PWM	priorita čítače/časovače 0 (1)
PX0, PX1	Pulse Width Modulation – pulzně šířková modulace
RAM	priorita vnějšího přerušení INT0 (INT1)
REP	paměť s libovolným přístupem
	registr pro opakování

RI, TI	příznak přerušení sériového kanálu
RXD, TXD	příjmový a vysílací signálový pin sériového kanálu
SBUF	vstupní nebo výstupní zásobník sériového kanálu
SCON	řízení sériové komunikace
SET	název tlačítka pro nastavení
SFR	speciální funkční registr
SP	ukazatel zásobníku
SPI	sériové periferní rozhraní
T0, T1	piny určené pro čítače
TF, TF0, TF1	příznakové bity přetečení čítače/časovače 0 (1)
TCON	Timer/Counter Control – registr pro nastavení čítače/časovače
TL0, TH0, TL1, TH1	dolní a horní registry čítače/časovače 0 (1)
TMOD	registr pro nastavení módu čítače/časovače 0 i 1
TR, TR0, TR1	kontrolní bit čítače/časovače 0 (1)
U _{CC} , GND	napájecí napětí a zem
UI	unipolární metoda

Použité symboly a veličiny

D	[%]	střída
f	[Hz]	frekvence
i, i _Z , I _Z	[A]	proud zátěže
I _{C_HA}	[A]	kolektorový proud tranzistoru T _{HA}
I _d	[A]	proud mezilehlého obvodu (ze zdroje do měniče)
I _{DO}	[A]	proud nulové diody
M	[Nm]	moment síly
t	[s]	čas
t ₀	[s]	ochranná doba
t _Z	[s]	doba zapnutí
t _V	[s]	doba vypnutí
T	[s]	perioda
u, u _Z , U _Z	[V]	napětí zátěže
U, U ₁	[V]	vstupní napětí
U _i	[V]	protinapětí
U _{STR}	[V]	střední hodnota napětí
u _P , U _P	[V]	napětí pily
U _{PMIN} , U _{PMAX}	[V]	minimální a maximální napětí pily
u _R , U _R	[V]	řídící napětí
u _{VA} , U _{VA}	[V]	větвовé napětí A
u _{VB} , U _{VB}	[V]	větвовé napětí B
z	[-]	zatěžovatel, poměrná doba zapnutí
ω	[rad/s]	úhlová rychlost

Seznam ilustrací a seznam tabulek

Obr. 1.1 Výukový systém	11
Obr. 1.2 Uspořádání pinů AT89S52	12
Obr. 1.3 Uspořádání programové paměti	14
Obr. 1.4 a) Uspořádání vnitřní datové paměti	15
b) Registr speciálních funkcí SFR	15
Obr. 1.5 Registry čítače/časovače 0 (1)	15
Obr. 1.6 Registr TMOD	15
Obr. 1.7 Registr TCON	16
Obr. 1.8 Registr IE	17
Obr. 1.9 Registr IP	18
Obr. 2.1 Výkonová jednotka	19
Obr. 2.2 PWM signál a výstupní napětí	20
Obr. 2.3 Pracovní režimy	21
Obr. 2.4 Pulzní měnič	
a) Pro snižování napětí	21
b) Pro zvyšování napětí	21
Obr. 2.5 Průběhy napětí a proudu na zátěži pulzního měniče	
a) Pro snižování napětí	21
b) Pro zvyšování napětí	21
Obr. 2.6 Čtyřkvadrantový pulzní měnič	22
Obr. 2.7 Průběh proudu v jednotlivých kvadrantech	
a) Průběh proudu v I. kvadrantu	23
b) Průběh proudu v II. kvadrantu	23
c) Průběh proudu v III. kvadrantu	23
d) Průběh proudu v IV. kvadrantu	23
Obr. 2.8 Princip řídicího systému čtyřkvadrantového PM – bipolární metoda	24
Obr. 2.9 Průběhy napětí v celém PM – bipolární metoda	25
Obr. 2.10 Princip řídicího systému čtyřkvadrantového PM – unipolární metoda	26
Obr. 2.11 Průběhy napětí v celém PM – unipolární metoda	27
Obr. 2.12 Průběhy v PM	
a) Bipolární metoda	28
b) Unipolární metoda	28
Obr. 3.1 Blokové schéma hlavní smyčky programu	29
Obr. 3.2 Blokové schéma 1. přerušení	30
Obr. 3.3 Blokové schéma 2. přerušení	32
Obr. 3.4 Tlačítka s jejich názvem a potenciometr	33
Obr. 3.5 Displej ukazující aktuální metodu	
a) Vypnuto	33
b) Bipolární	33
c) Unipolární	33
Obr. 3.6 Displej ukazující aktuální frekvenci	34
Obr. 3.7 Signalizace LED diod a jejich význam	34

Obr. 4.1 Řídicí signály při bipolárním řízení s frekvencí 70Hz a střídou 20%.....	35
Obr. 4.2 Řídicí signály při bipolárním řízení s frekvencí 130Hz a střídou 70%.....	36
Obr. 4.3 Řídicí signály při unipolárním řízení s frekvencí 70Hz a střídou 20%.....	36
Obr. 4.4 Řídicí signály při unipolárním řízení s frekvencí 130Hz a střídou 70%.....	37
Obr. 4.5 Napětí a proud na odporové zátěži při bipolárním řízení s frekvencí 70Hz a střídou 20%	38
Obr. 4.6 Napětí a proud na odporové zátěži při bipolárním řízení s frekvencí 130Hz a střídou 70%	38
Obr. 4.7 Napětí a proud na odporové zátěži při unipolárním řízení s frekvencí 70Hz a střídou 20%	39
Obr. 4.8 Napětí a proud na odporové zátěži při unipolárním řízení s frekvencí 130Hz a střídou 70%	39
Obr. 4.9 Napětí a proud na motorické zátěži při bipolárním řízení s frekvencí 70Hz a střídou 20%	40
Obr. 4.10 Napětí a proud na motorické zátěži při bipolárním řízení s frekvencí 130Hz a střídou 70%	40
Obr. 4.11 Napětí a proud na motorické zátěži při unipolárním řízení s frekvencí 70Hz a střídou 20%	41
Obr. 4.12 Napětí a proud na motorické zátěži při unipolárním řízení s frekvencí 130Hz a střídou 20%	41
Obr. 4.13 Napětí a proud na motorické zátěži při unipolárním řízení s frekvencí 130Hz a střídou 70%	42
Tab. 1 Vektory přerušení	17
Tab. 2 Popis tlačítek	33
Tab. 3 Naměřené hodnoty řídicích signálů – bipolární metoda.....	37
Tab. 4 Naměřené hodnoty řídicích signálů – unipolární metoda.....	37

Obsah

Zadání bakalářské práce	2
Prohlášení	3
Poděkování	3
Abstrakt	4
Klíčová slova	4
Seznam použitých symbolů a značek	5
Seznam ilustrací a seznam tabulek	7
Obsah	9
Úvod	10
1. Výukový systém a mikrokontrolér	11
1.1 Popis výukového systému a připojených periférií	11
1.2 Základní specifikace mikrokontroléru a porty	12
1.3 Paměti	13
1.4 Čítače a časovače	15
1.5 Přerušení	16
2. Výkonová část	19
2.1 Popis	19
2.2 Pulzně šířková modulace	20
2.3 Pulzní měnič	21
2.4 Bipolární metoda	24
2.5 Unipolární metoda	26
2.6 Porovnání průběhů jednotlivých metod	28
3. Software	29
3.1 Diagram programu	29
3.2 Ovládání programu	33
4. Průběhy a ověření	35
Závěr	43
Použitá literatura	44
Seznam příloh	45

Úvod

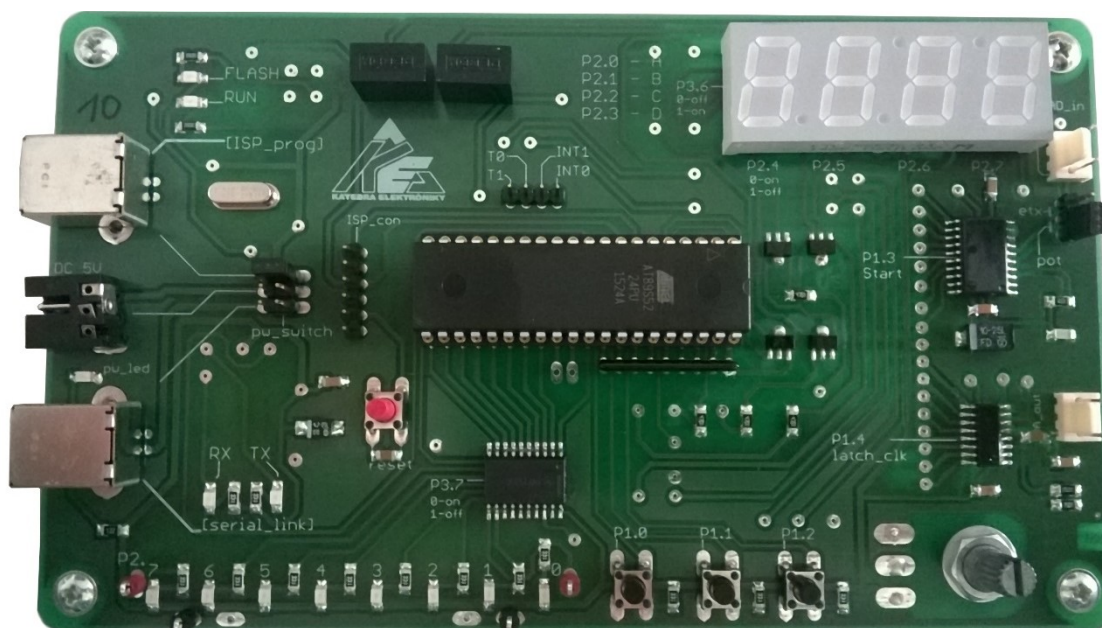
V této bakalářské práci lze vidět, že i pouhý mikroprocesor řady 8051, je schopný ovládat výkonovou jednotku s pulzním měničem. Stejnoseměrné pohony jsou v dnešní době velmi rozšířené a použití je možné najít například v tramvajích, lokomotivách, ale také v robotech. A tyto pohony je nutno regulovat. Řídí se zejména napětí, čímž se řídí rychlost otáček. Velmi jednoduše je toho docíleno právě pulzním měničem, do kterého se posílají řídicí signály z mikroprocesoru. Metody pro řízení pulzního měniče máme dvě: bipolární a unipolární, které budou později podrobněji popsány. Zaznamenám také reálné průběhy, jednak vstupních signálů, ale také výstupních veličin. Následně tento software bude sloužit pro demonstrační účely v laboratorních cvičeních.

V první kapitole této bakalářské práce bude popsán výukový systém, co všechno obsahuje, kde jsou dané periferie připojeny a hlavně mikroprocesor, který se na přípravku nachází. Následně bude popsána výkonová část, a jak již bylo zmíněno dříve, metody řízení. Důležitou součástí této práce je software pro ověření řízení, jeho popis a ovládání. Nakonec zobrazení funkčnosti softwaru.

1. Výukový systém a mikrokontrolér

1.1 Popis výukového systému a připojených periférií

Výukový systém obsahuje řadu periférií, a to: tlačítka, LED diody, 7 segmentový displej, D/A převodník, A/D převodník na který je připojen potenciometr, ale je možné přepojit převodník na externí vstup, piny pro vnější přerušení a konektory pro napájení a přeprogramování mikroprocesoru. Samozřejmě zde najdeme mikrokontrolér AT89S52, který vše řídí.



Obr. 1.1 Výukový systém

LED diody jsou zapojeny na port P2. Jelikož není dovoleno na mikrokontroléru mít vysoké zatížení výstupu, tak jsou buzeny proti napájecímu napětí. Pro tento účel byl vybrán sběrnicový posilovač 74AC245. Aby se LED diody rozsvítily, je potřeba přivést logickou 0 na vstup OE(G) připojen na pin P3.7, kterým se zapne budič a vstupy A1 – A8 přejdou na výstupy B1 – B8 na kterém jsou zapojeny LED diody. Pochopitelně je nutné vyslat datové slovo na port P2 podle toho, jak je žádané, aby svítily, popřípadě nesvítily dané LED diody.

Displej je zapojen přes BCD dekodér. Tento dekodér převádí 4 bity, binární hodnotu, na jednotlivé segmenty pro displej. Taktéž je i tento dekodér na sběrnici P2 a další 4 vodiče jsou připojeny do báze tranzistorů, které určují jednotlivé pozice na displeji. Takže pokud se má rozsvítit celý displej (všechny 4 pozice), je potřeba jej multiplexovat. To znamená, že je potřeba rozsvěcovat postupně jednotlivé pozice, k čemuž slouží právě tranzistory. Pomocí multiplexování lze omezit množství vodičů a také ovládacích prvků. Pro zde uvedený displej by bylo zapotřebí 20 vodičů a 4 dekodéry, na místo 8 vodičů a pouze 1 dekodéru. Multiplexování (přepínání) daných pozic musí probíhat dostatečně rychle, aby lidské oko nedokázalo zaznamenat, že svítí jen jedna pozice.

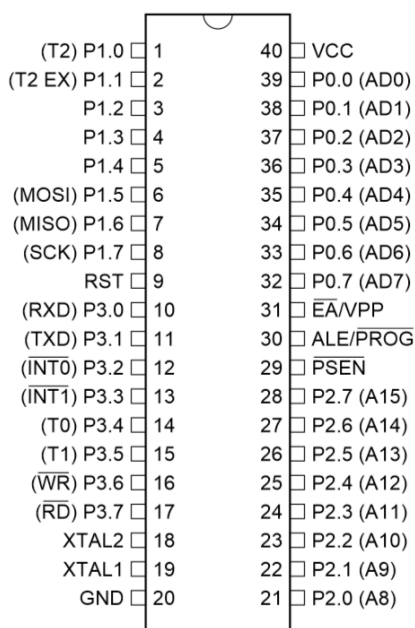
1. Výukový systém a mikrokontrolér

D/A převodník slouží k převodu vstupního datového slova na výstupní analogový signál. Použití převodníku se využívá při generování průběhů, nebo např. řízení dalších členů připojených na převodník. Tento převodník je opět připojen na bránu P2 a jelikož je zde již připojen displej a LED diody, je před ním ještě záchytný registr. Proto je potřeba poslat na vstup CLK signál, který uloží datové slovo a pošle na výstup. Vstup CLK je připojen na pin P1.4.

A/D převodník je opakem D/A převodníku. To znamená, že převádí analogovou veličinu na digitální (datové slovo). Využití se nachází například v audio technice, kde se převádí signál uložený v digitální podobě na analogový signál. A/D převodník je připojen na bránu P0 a pro začátek převodu je potřeba přivést log. 1 na pin P1.3. Pokud není zapotřebí, aby převodník převáděl, stačí přivést log. 0 na pin P1.3. Na převodník je připojen potenciometr, ale lze také snímat signál z externího zdroje a to tak, že se přehodí jumper do druhé pozice a připojí se signál.

1.2 Základní specifikace mikrokontroléru a porty

Mikrokontrolér je integrovaný obvod, nebo čip s procesorem spolu s dalšími podpůrnými zařízeními, jako jsou programové paměti, datové paměti, vstupně/výstupní porty, sériové komunikační rozhraní a další zařízení integrovány dohromady. Na rozdíl od mikroprocesoru již mikrokontrolér nevyžaduje žádné další periferie, aby fungoval. Mikrokontrolér AT89S52, který se nachází na výukovém systému je 8 bitový, obsahuje 4kB paměti typu Flash (programová paměť) a je možné ji rozšířit vnější pamětí na 64kB. Mikrokontrolér má 2 vstupy pro napájecí napětí - U_{CC} a GND, 3 systémové vstupy – jeden pro reset a další dva pro krystal, 3 řídicí signály EA (External Access – slouží pro výběr čtení programu z vnitřní nebo vnější paměti), PSEN (Program Store Enable – používá se jako signál při čtení programu z externí paměti), ALE (Adress Latch Enable – používá se pro mikroprocesor s 16b adresovou sběrnicí pro multiplexování) a 4 porty pro vnější komunikaci.



Obr. 1.2 Uspořádání pinů AT89S52

Vstupní a výstupní porty slouží pro připojení mikropočítače k dalším periferiím, jako je klávesnice, displej, LED diody, akční členy a mnoho dalších. Mikrokontrolér má 4 porty, z nichž každý má 8 bitů.

Port 0 kromě toho, že slouží jako vstupní nebo výstupní port, tak se používá pro signály adresových a datových sběrnic nižšího řádu a jsou multiplexovány právě tímto portem. Jejich účel je pro vnější paměťové rozhraní. Jedná se o obousměrný port (jediný u 8051) a ke své funkci vyžaduje pull-up (zdvihací) rezistory.

Port 1 na rozdíl od ostatních portů nemá žádné další funkce. Používá se zejména pro vnější periferie jako LCD displej, klávesnice aj. Avšak oproti předcházejícímu portu, tento již disponuje vnitřními pull-up rezistory, čímž se určí logická hodnota ve chvíli, kdy je nedefinována. To může nastat například, když na vstupu není připojena žádná periferie. I když celý port nemá další funkce, tak některé piny jej mají. Například P1.5 až P1.7 slouží pro SPI.

Port 2 stejně jako port 0 má více funkcí. Dá se použít jako přístup k jiným zařízením, ale také je možné jej využít pro adresy externí paměti, datové nebo programové. Stejně jako port 1 je tento port vybaven pull-up rezistory.

Port 3 je vybaven pull-up rezistory a celý port má alternativní funkce. To znamená, že každý pin má kromě vstupně výstupní funkce, taky svou vlastní funkci. První dva piny se používají k sériové komunikaci. P3.0 označen jako RXD je příjmový signálový pin a používá se pro příjem dat ze sériové komunikace. P3.1 označen jako TXD je vysílací signálový pin a používá se pro vysílání dat po sériové komunikaci. Další dva piny slouží pro přerušení. P3.2 označen jako INT0 je externí přerušení 0 a pin P3.3, který je označen jako INT1 je externí přerušení 1. Pokud je dané přerušení povoleno a dojde například ke stisku tlačítka, které je připojeno právě na tento pin, tak se vykoná dané přerušení a poté se program vrátí zpět do hlavní smyčky. Následné dva piny jsou určeny pro čítače. P3.4 je označen T0 a P3.5 jako T1. Pokud je správně nakonfigurovaný čítač, tak tyto piny budou sloužit pro inkrementaci čítače pokaždé, když se detekuje přechod z log. 1 na log. 0. Poslední dva piny jsou určeny pro čtení a zápis z vnější paměti.

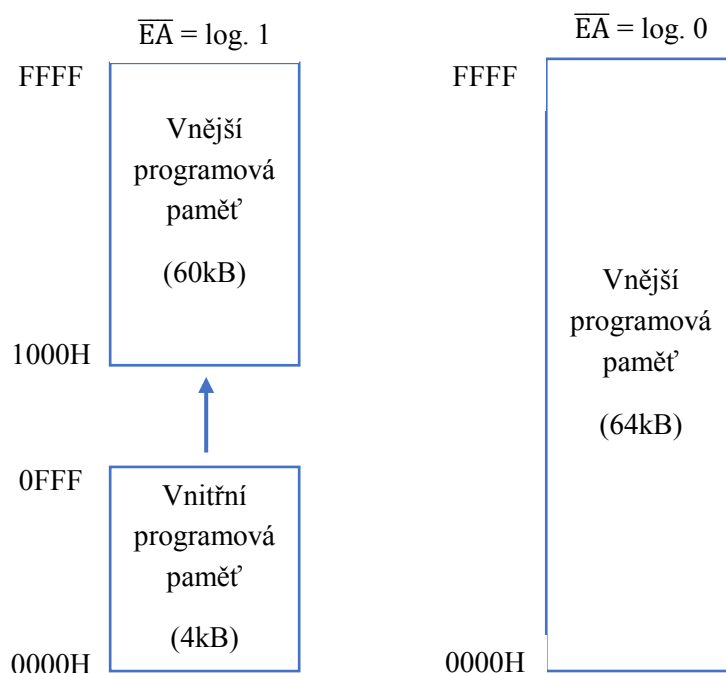
1.3 Paměti

Předtím než půjde hlouběji, je nutné si povědět něco málo o architektuře. U mikropočítače se tím myslí organizace, struktura, funkce a realizace mikropočítače. Rozlišují se dvě architektury: Harvardská architektura a architektura Von Neumann. Hlavní rozdíl mezi těmito architekturami je to, že u Harvardské je paměť pro program a data oddělená, zatímco u architektury Von Neumann je společná. Obě architektury mají svoje kladné stránky a také záporné.

Za kladné stránky u Harvardské architektury se považuje to, že nemůže dojít k přepsání programu daty. Datová a programová paměť má svou vlastní sběrnici, přes kterou je připojena k aritmeticko-logické jednotce. Přenos dat a instrukcí je možný zároveň, což má za následek větší rychlost. Naopak nevýhodou je mapování paměti, to znamená, že již při návrhu je nutné vědět, kolik bude zapotřebí místa pro data a kolik pro program.

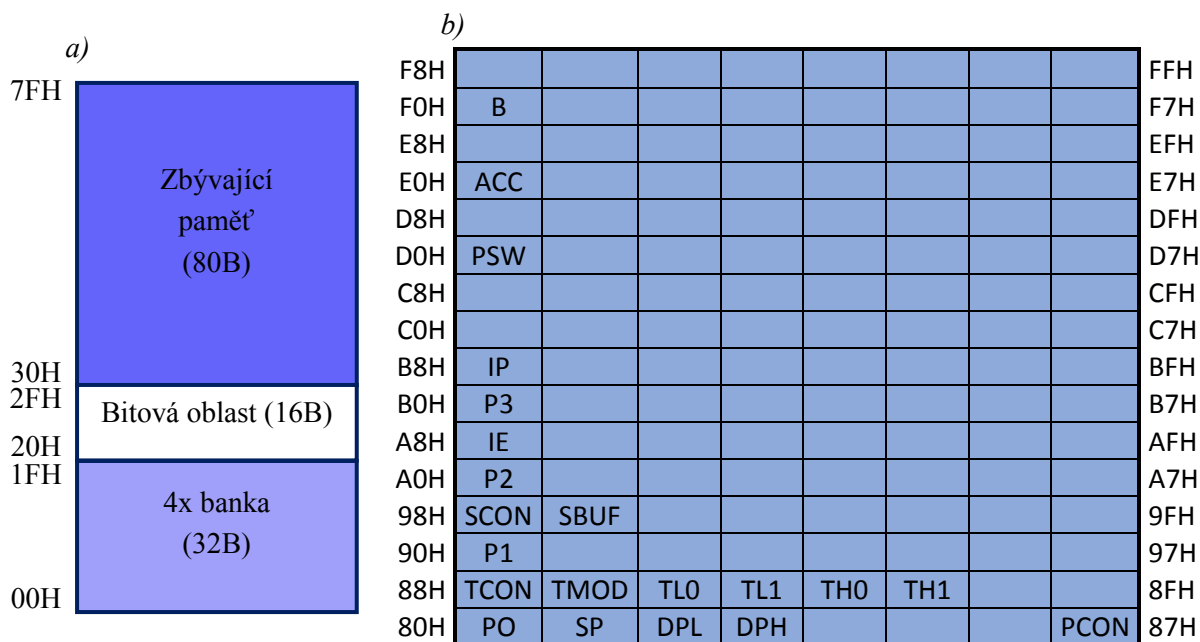
Naopak u architektury Von Neumann lze měnit místo v paměti plynule, ale za to může dojít k tomu, že data přepíší program, popřípadě program přepíše data. Jelikož je paměť společná, tak přenos dat a instrukcí nemůže probíhat současně a tím je proces pomalejší.

Mikroprocesory 8051 jsou Harvardské architektury, tedy jejich paměti jsou oddělené. Přístup k těmto pamětím je možný pomocí různých instrukcí. Mikroprocesor obsahuje 4kB programové paměti a v případě potřeby je možné ji rozšířit vnější pamětí na 64kB. Ve výchozím nastavení by měl být pin EA připojen na log. 1 (napájecí napětí U_{CC}). To má za následek, že instrukce se budou číst z vnitřní paměti a pokud dojde k překročení limitu paměti 4kB, tak se automaticky přesune na vnější paměť, odkud se budou číst instrukce. Pokud je ale vyžadováno číst instrukce pouze z vnější paměti tak, že se přeskočí vnitřní paměť, je nutné připojit pin EA na log. 0 (GND). 8051 má speciální schopnost uzamknout programovací paměť, čímž se zabrání softwarovému pirátství. Tato funkce je možná aktivovat pomocí lock (uzamykacích) bitů. Jakmile jsou tyto bity naprogramovány, tak již nelze přistupovat do vnitřní paměti použitím vnějších obvodů. Nicméně uzamknutí není možné, pokud vnější paměť je také použita pro uložení programového kódu. Jakmile již jsou bity uzamknuty, tak je lze odemknout pouze smazáním spolu s programem z vnitřní paměti.



Obr. 1.3 Uspořádání programové paměti

Datová paměť v 8051 má 128 bytů a lze ji rozšířit vnější pamětí o 64kB. Celková datová paměť může tedy být 64kB (vnější) + 128B (vnitřní). Datovou paměť tvoří 3 části: nejprve obsahuje 4 banky, kde každou banku tvoří osm registrů a jsou značené R0 až R7. Poté je bitová oblast, pro kterou je vyhrazeno 16 bytů a nachází se na adresách 20H až 27H, využívá se například jako stav LED diody, nebo motoru zapnuto/vypnuto. Následně už je zbývajících paměť v oblasti 30H až 7FH. Na vyšších adresách, 80H až FFH se nachází registr speciálních funkcí SFR. Jedna oblast v registru speciálních funkcí má 1 byte. Na obrázku jsou zobrazeny adresy v hexadecimální soustavě.



Obr. 1.4 a) Uspořádání vnitřní datové paměti b) Registr speciálních funkcí SFR

1.4 Čítače a časovače

Čítače a časovače jsou nedílnou součástí mikroprocesoru a jejich funkce je velmi prostá. Čítač i časovač se stará o paměťové místo, a když zjistí, že došlo k vzestupné nebo sestupné hraně signálu, záleží podle nastavení, tak přičte jedničku. Čítač čítá impulsy od vnějšího signálu, například z čidla, zatímco časovač čítá impulsy od vnitřního signálu se stálým kmitočtem, takzvaného hodinového signálu. Používá se zejména pro zpožďovací smyčku, ale může být použit k vytvoření časového intervalu. Mikroprocesor 8051 má dva čítače/časovače a můžou být použity libovolně: dva čítače, nebo jeden čítač a druhý časovač, popřípadě oba jako časovače. Oba čítače/časovače mají 16 bitů, i když je mikroprocesor 8051 pouze 8 bitové architektury. Proto je registr rozdělen na nižší a vyšší byte. Tyto registry jsou pojmenovány TL0 a TH0 pro čítač/časovač 0 a další registry TL1 a TH1 pro čítač/časovač 1.

TH0 (TH1)								TL0 (TL1)							
D15	D14	D13	D12	D11	D10	D9	D8	D7	D6	D5	D4	D3	D2	D1	D0

Obr. 1.5 Registry čítače/časovače 0 (1)

Jeden z důležitých registrů pro nastavení čítače/časovače je takzvaný TMOD (Timer MODE) registr a jeho adresa v paměti RAM je 89H. Tento registr je 8 bitový a používá se pro nastavení obou čítačů a časovačů. Nižší 4 bity slouží pro nastavení čítače/časovače 0 a vyšší 4 bity pro čítač/časovač 1. Na následujícím obrázku můžeme vidět celý registr a jednotlivé bity.

Čítač/časovač 1				Čítač/časovač 0			
GATE	C/ \bar{T}	M1	M0	GATE	C/ \bar{T}	M1	M0

Obr. 1.6 Registr TMOD

GATE - slouží k určení vnějšího nebo vnitřního zdroje pro spuštění a vypnutí čítače/časovače
 - při log. 0 je ovládán vnitřním bitem TR
 - při log. 1 je ovládán vnitřním bitem TR a vnějším bitem INT

C/T - slouží k určení režimu čítač nebo časovač
 - při log. 0 bude pracovat jako časovač
 - při log. 1 bude pracovat jako čítač

M1, M0 - slouží k určení módu čítače/časovače
 - mód 0 (M1 = 0, M0 = 0) - pracuje jako 13 bitový čítač/časovač, registr TH má 8b a TL má 5b
 - mód 1 (M1 = 0, M0 = 1) - pracuje jako 16 bitový čítač/časovač, registr TH i TL mají 8b
 - mód 2 (M1 = 1, M0 = 0) - pracuje jako 8 bitový čítač/časovač, po přetečení se automaticky nastaví do registru TL hodnota z registru TH
 - mód 3 (M1 = 1, M0 = 1) - slouží k časování komunikace pro sériový kanál

Při přetečení čítače (stav, kdy ze samých jedniček přejde čítač/časovač na samé nuly) dojde k nastavení takzvaného příznakového bitu TF, který se nachází v registru TCON. Tento bit může být využit jako zdroj přerušení.

Další důležitý registr, který již byl zmíněný je TCON (Timer/Counter Control) a jeho adresa v paměti RAM je 88H. Vidět tento registr je možné na následujícím obrázku.

TF1	TR1	TF0	TR0	IE1	IT1	IE0	IT0
-----	-----	-----	-----	-----	-----	-----	-----

Obr. 1.7 Registr TCON

TF0, TF1 - příznakový bit, který se nastaví při přetečení čítače/časovače a sám se taky vymaže, když dojde do příslušného podprogramu přerušení

TR0, TR1 - kontrolní bit, který slouží ke spuštění nebo zastavení daného čítače/časovače

IE0, IE1 - příznakový bit vnějšího přerušení, daný bit se nastaví buď při sestupné hraně, nebo log. 0 na pinu INT0 (INT1) podle nastavení bitu IT0 (IT1). Když dojde do příslušného podprogramu přerušení, tak se bit sám vynuluje

IT0, IT1 - nastavení přerušení na sestupnou hranu signálu nebo log. 0 na pinu INT0 (INT1)

1.5 Přerušení

Nejdůležitější a největší roli v mikroprocesorech má přerušení. Funkce přerušení je velmi jednoduchá. Mikroprocesor vykonává hlavní smyčku, a když dojde k požadavku o přerušení, hlavní smyčka se pozastaví, uloží se adresa poslední provedené instrukce a skočí se do obsluhy přerušení. Po dokončení obsluhy přerušení se mikroprocesor navrátí zpět do hlavní smyčky na místo, kde skončil, respektive na adresu, kterou si uložil. To vše za předpokladu, že je přerušení správně nastavené a povoleno. Je možné si uvést příklad. V hlavní smyčce probíhá sekvence probíhání LED diod, v tom

1. Výukový systém a mikrokontrolér

okamžiku se stiskne tlačítko, to vyvolá přerušení, v kterém se změní například směr problikávání. V obsluze přerušení se tedy přepíše konstanta pro směr. Jakmile je obsluha dokončena, tak se mikroprocesor vrátí zpět do hlavní smyčky a opět pokračuje tam, kde skončil, tj. problikáváním LED diod, ovšem teď v jiném směru. Další využití lze najít, když je potřeba čekat například jednu vteřinu. Pokud by se v hlavní smyčce udělal cyklus, kde by se musel čekat tento čas, tak by mikroprocesor v tu chvíli nedělal nic jiného, než přičítal nebo odečítal jedničku, až by uplynula jedna vteřina. Proto se využije časovač, který je popsán o kapitulu výše. V hlavní smyčce se teď může dělat něco užitečného a v momentě, kdy uplyne požadovaný čas, tak dojde k přerušení a následné obsluze.

Mikroprocesory mají několik zdrojů přerušení a záleží na daném typu mikroprocesoru, kolik jich mají. U mikroprocesoru 8051 je jich přesně pět: dva vnější přerušení, dva přerušení od čítače/časovače a přerušení od sériového kanálu. Každé přerušení má svůj příznak a svoji adresu, na kterou skočí, když dojde k požadavku o přerušení.

Tab. 1 Vektory přerušení

Adresa	Typ přerušení	Příznak
0003H	Vnější přerušení 0	IE0
000BH	Čítač / časovač 0	TF0
0013H	Vnější přerušení 1	IE1
001BH	Čítač / časovač 1	TF1
0023H	Sériový kanál	RI + TI

Při resetu mikroprocesoru 8051 se veškeré přerušení zakážou. Aby bylo možné použít některé z těchto přerušení, je potřeba dané přerušení povolit. K tomu slouží registr IE (Interrupt Enabled). Tento registr se nachází na adrese A8H v paměti RAM.

EA	-	-	ES	ET1	EX1	ET0	EX0
----	---	---	----	-----	-----	-----	-----

Obr. 1.8 Registr IE

- EA - Enable All, slouží k povolení všech přerušení
- ES - určený pro sériový kanál, povolení přerušení od vysílání a příjmu sériového kanálu
- ET0, ET1 - používá se pro povolení přerušení čítače/časovače 0 (1), způsobené přetečením
- EX0, EX1 - povolení vnějšího přerušení INT0 (INT1)

1. Výukový systém a mikrokontrolér

Registr IE je nastavený a je povoleno pouze přerušení, které je zapotřebí. Ale pokud jich je více, nastává otázka, co se bude dít, když dojde k přerušení z více zdrojů ve stejnou dobu. Které přerušení bude obsloužené jako první, určí systém priority. Mikroprocesor má systém priority řešený tak, že jednotlivé přerušení jsou pevně přiřazeny prioritou v každé ze dvou programově volitelných úrovní. A ty jsou označeny jako vyšší a nižší. Nejvyšší prioritu má vnější přerušení 0, za ním je další přerušení od čítače/časovače 0, následně je vnější přerušení 1, poté čítač/časovač 1 a jako poslední je přerušení od sériového kanálu. Někdy je tohle pevné přidělení priorit nazýváno jako přirozená priorita. Zásadou pevného stanovení priorit je jasné, jaké bude pořadí obsloužení přerušení. Avšak ne vždy nám dané pořadí vyhovuje. Právě proto jsou dvě úrovně priorit. Pokud by bylo zapotřebí změnit prioritu přerušení, může se toho docílit registrem IP, který se nachází na adrese B8H v paměti RAM. V tomto registru se zvolí přerušení, která mají mít vyšší úroveň a ty budou obslouženy jako první. Avšak i zde platí pevné stanovení priority. To znamená, že pokud se do vyšší úrovně přepne vnější přerušení 1 a čítač/časovač 1, tak se nejprve vykoná přerušení vnější a za ním čítač/časovač 1. Poté se dostanou na řadu přerušení s nižší úrovní zase podle pevné priority.

-	-	-	PS	PT1	PX1	PT0	PX0
---	---	---	----	-----	-----	-----	-----

Obr. 1.9 Registr IP

PT0 - priorita čítače/časovače 0

PX0 - priorita vnějšího přerušení INT0

PT1 - priorita čítače/časovače 1

PX1 - priorita vnějšího přerušení INT1

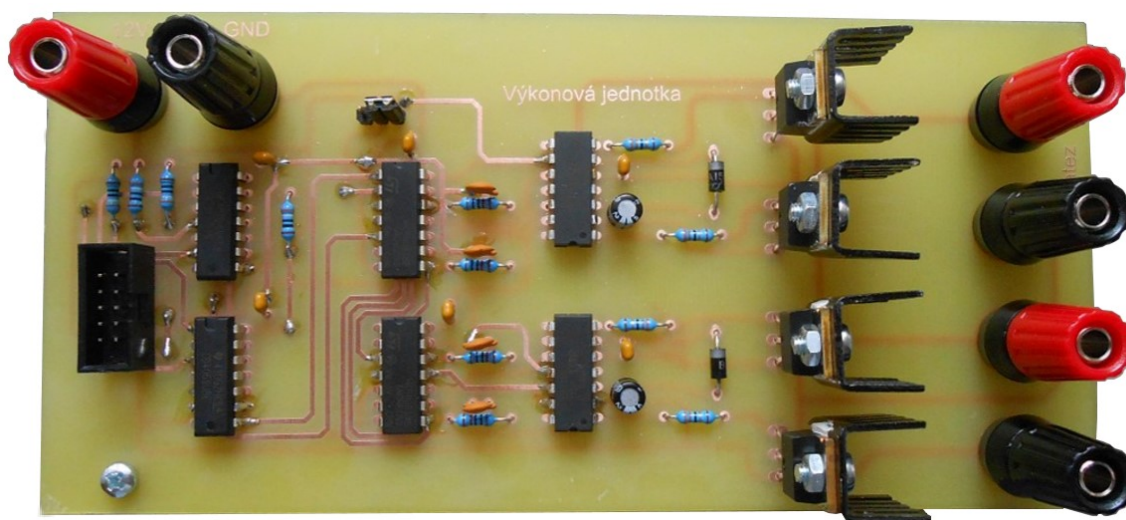
PS - priorita sériového kanálu

Nastavením log. 0 se přesune dané přerušení do nižší priority a naopak nastavením log. 1 se přesune dané přerušení do vyšší priority.

2. Výkonová část

2.1 Popis

Výkonová část obsahuje šest integrovaných obvodů, několik pasivních součástek, a to nejdůležitější čtyři výkonové tranzistory. Na výkonové jednotce se nachází několik konektorů. Konektory označeny 12V a GND slouží pro přívod napájecího napětí pro všechny integrované obvody, kromě jednoho. Ten je napájen 5V z konektorové patice a to proto, že výstupní napětí z mikroprocesoru je právě 5V. Za integrovaným obvodem již dojde ke zvýšení napětí na úroveň 12V. Další konektory označeny jako OUT+ a OUT- slouží pro napájení pulzního měniče. Poslední dva konektory s názvem zátěž slouží pro připojení zátěže.



Obr. 2.1 Výkonová jednotka

Na výkonové jednotce je možné si povšimnout jednoho jumpru. Ten slouží pro trvalé vypnutí pulzního měniče. Při resetu mikroprocesoru se na výstupech objeví log. 1, čímž by došlo k sepnutí všech tranzistorů. Aby k tomu nedošlo, tak je na výkonové jednotce invertor, který neguje vstupy. To znamená, že k sepnutí tranzistoru dochází, pokud je na vstupu log. 0.

Při přechodu ze zapnutého stavu tranzistoru do vypnutého a naopak je nutné zajistit takzvanou ochrannou dobu. To proto, aby nedošlo ke stavu, kdy budou oba tranzistory v jedné větvi sepnuty, jinak by došlo ke zkratu a následnému zničení spínacích prvků. Ochranná doba je provedena pomocí pasivních součástek RC a integrovaným obvodem typu AND (logický součin). Takže pokud přijde signál od mikroprocesoru pro sepnutí tranzistoru, tak se na jednom vstupu objeví log. 1 okamžitě, zatímco na druhý vstup přijde až po době, kdy se kondenzátor nabije přes odpor a až tehdy se na výstupu objeví také log. 1. Ale pro rozepnutí tranzistoru se na vstupu objeví log. 0 a ta okamžitě i na výstupu. Tudíž při návrhu softwaru již nemusíme s ochrannou dobou počítat.

2.2 Pulzně šířková modulace

Pulzně šířková modulace, zkráceně jen PWM (Pulse Width Modulation), je metoda pro generování analogového signálu při použití digitálního zdroje. PWM signál se skládá ze dvou hlavních složek: střída a frekvence. Střída určuje, jak dlouho je signál ve stavu log. 1 vzhledem k jednomu cyklu a uvádí se v procentech. Frekvence nám říká, kolik cyklů se provede za jednotku času, například 100Hz by bylo 100 cyklů za vteřinu. Pokud se bude zapínat a vypínat digitální signál dostatečně rychle s určitou střídou, tak se výstup bude jevit jako konstantní analogové napětí. To se využívá například pro regulaci jasu LED diod, v současnosti je však hodně využívána u měničů. Střední hodnota napětí se potom vypočte jako:

$$U_{STŘ} = \frac{1}{T} \int_0^T u(t) dt$$

Jelikož napětí může nabývat pouze dvou hodnot 0V při log. 0 a U_1 při log. 1, tak se ví, že se jedná o obdélníkový průběh. V tom případě se může střední hodnotu napětí vypočítat jako:

$$U_{STŘ} = \frac{1}{T} \int_0^T u(t) dt = \frac{1}{T} \left(\int_0^{t_z} U_1 dt + \int_{t_z}^T 0 dt \right) = \frac{1}{T} \cdot U_1 \cdot [t]_0^{t_z} = \frac{1}{T} \cdot U_1 \cdot (t_z - 0) = U_1 \cdot \frac{t_z}{T} = U_1 \cdot z$$

Kde: U_1 - vstupní napětí

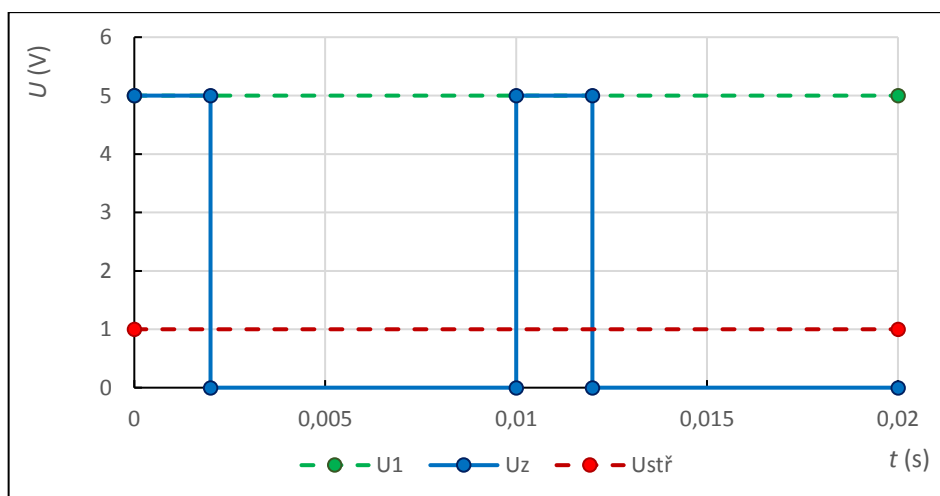
t_z - doba zapnutí

T - perioda

z - zatěžovatel, poměrná doba zapnutí

Příklad: Vstupní napětí $U_1 = 5V$, doba zapnutí $t_z = 2ms$, perioda $T = 10ms$.

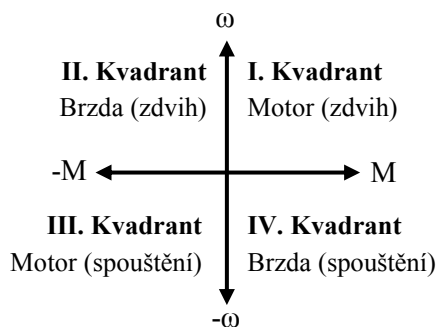
$$U_{STŘ} = U_1 \cdot \frac{t_z}{T} = 5 \cdot \frac{2 \cdot 10^{-3}}{10 \cdot 10^{-3}} = 1V \quad D = \frac{t_z}{T} \cdot 100 = \frac{2 \cdot 10^{-3}}{10 \cdot 10^{-3}} \cdot 100 = 20\%$$



Obr. 2.2 PWM signál a výstupní napětí

2.3 Pulzní měnič

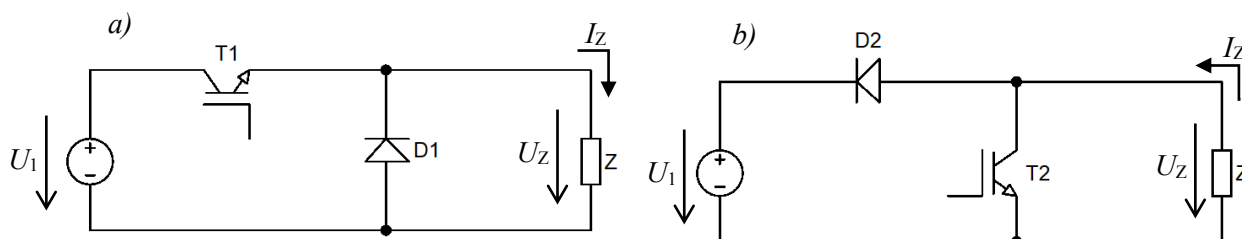
Pro řízení otáček motoru je potřeba měnit velikost střední hodnoty napětí. K tomu se využívá právě pulzní měnič. Pokud je požadováno řídit motor, je důležité si uvědomit, pro jaký režim bude pulzní měnič určen. Rozeznává se motorický a generátorický režim a také jakým směrem se hřídel otáčí. Proto jsou čtyři kvadranty. Typickým příkladem pohonu, který pracuje ve čtyřech kvadrantech, je výtah. Jednotlivé kvadranty je možné vidět na následujícím obrázku.



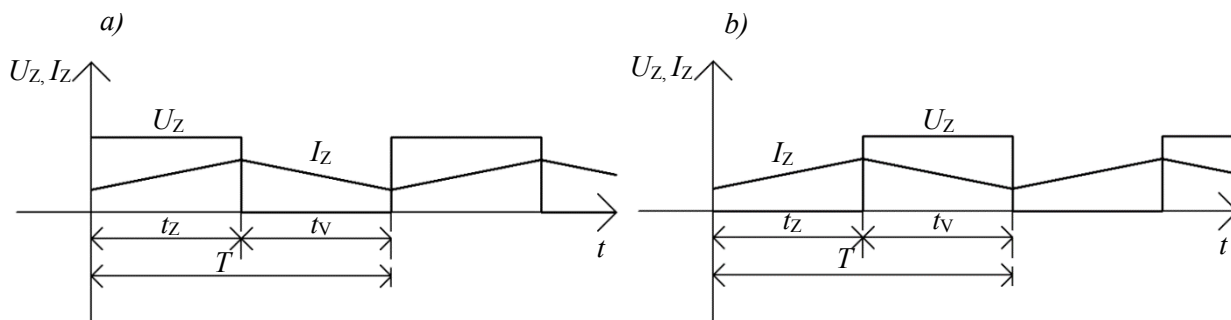
Obr. 2.3 Pracovní režimy

Jak je patrné z obrázku, motorický režim odpovídá I. a III. režimu při zdvihu a spouštění, zatímco II. a IV. kvadrant potom náleží generátorickému režimu při zdvihu a spouštění.

Nejjednodušší pulzní měnič je realizován pomocí jediného spínacího prvku a pracuje pouze v jednom kvadrantu. Může se jednat o pulzní měnič pro snižování napětí nebo pro zvyšování napětí. Zapojení je zobrazeno na následujících obrázcích. Pro správnou funkčnost PM pro zvyšování napětí musí měnič obsahovat zdroj protinapětí U_i .



Obr. 2.4 Pulzní měnič a) Pro snižování napětí b) Pro zvyšování napětí



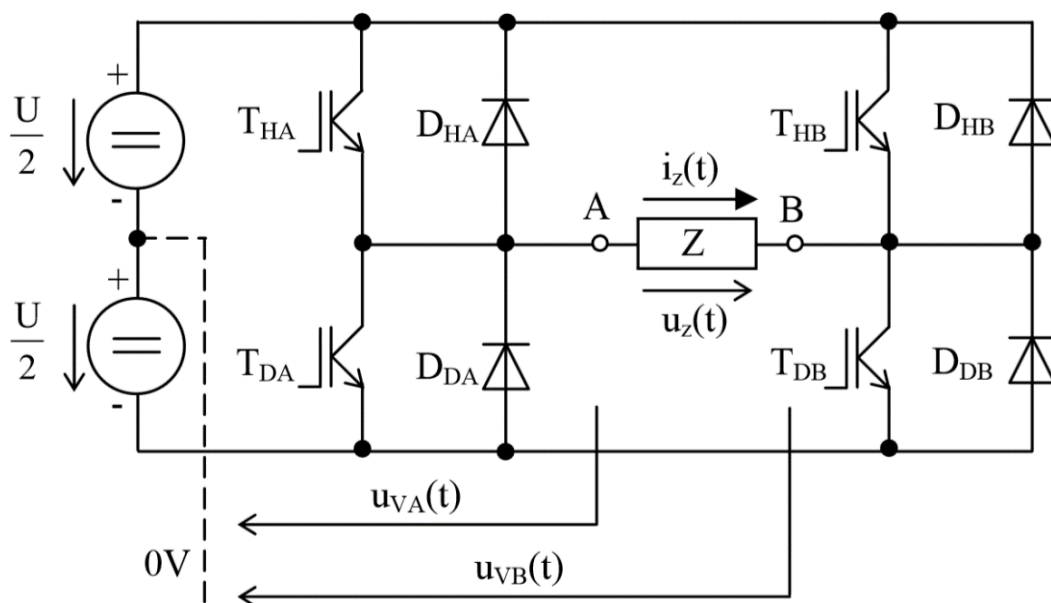
Obr. 2.5 Průběhy napětí a proudu na zátěži pulzního měniče
a) Pro snižování napětí b) Pro zvyšování napětí

Z obrázku 2.4a je možné vidět, že PM pro snižování napětí je tvořen spínačem T1 a nulovou diodou D1. Takovéto zapojení slouží k řízení napětí na zátěži Z. Napětí na výstupu je menší než napětí napájecího zdroje U_1 , proto se nazývá PM pro snižování napětí. Z průběhu napětí a proudu PM pro snižování napětí je patrné, že při sepnutí tranzistoru T1 se na zátěži objeví vstupní napětí U_1 a při rozepnutí nulové napětí. To platí pro spojitý proud, kde proud má průběh podle typu zátěže. V případě odporové zátěže by proud kopíroval napětí, indukční zátěž by měla pilový průběh a RL zátěž exponenciální. Při sepnutí by samozřejmě proud rostl a při rozepnutí klesal. Když je tranzistor sepnut, tak proud teče přes něj, ale v momentě rozepnutí tranzistoru se proud uzavírá přes diodu. Střední hodnota napětí se určí stejně jako u pulzně šířkové modulace.

$$U_{STŘ} = U_1 \cdot \frac{t_z}{T} = U_1 \cdot z$$

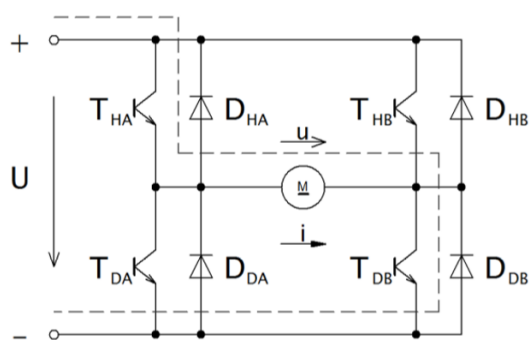
Někdy je zapotřebí převést energii z nižšího zdroje napětí do vyššího zdroje napětí. Například při generátorickém brzdění stejnosměrného motoru. V motorickém režimu motor čerpá energii z baterie, ale pokud je zapotřebí zabrzdít, tak je kotevní napětí motoru U_i menší než na baterii. A v této situaci se využije měnič pro zvyšování napětí. Na obrázku 2.4b je opět možné vidět zapojení, které je tvořeno spínačem T2 a oddělovací diodou D2. U pulzního měniče pro zvyšování napětí je k vidění, že při sepnutém tranzistoru T2 je napětí nulové. Zátěž je totiž ve zkratu. Dioda D2 však zabráňuje zkrat zdroje. Proud se uzavírá přes tranzistor T2 a roste. Naopak při rozepnutí tranzistoru proud teče diodou. Střední hodnota napětí se vypočte podle vzorce:

$$U_{STŘ} = U_1 \cdot \frac{t_v}{T} = U_1 \cdot \frac{T - t_z}{T} = U_1 \cdot (1 - z)$$

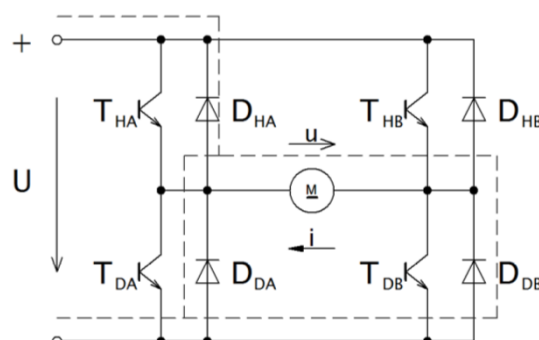


Obr. 2.6 Čtyřkvadrantový pulzní měnič

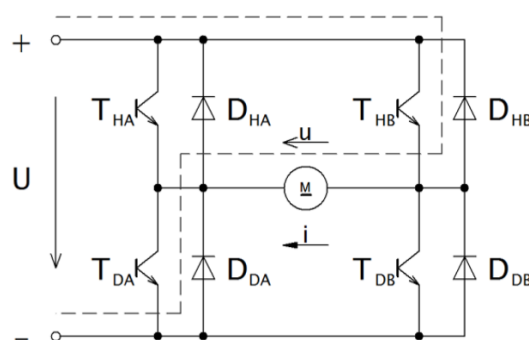
Schéma zapojení pulzního měniče uvedené na obr. 2.6 je schopné pracovat ve všech čtyřech kvadrantech. Tranzistory a diody, které jsou sepnuty, udávají kvadrant, ve kterém se pulzní měnič právě nachází. V prvním kvadrantu, kde napětí a proud je kladný, jsou aktivní tranzistory T_{HA} a T_{DB} . Nacházíme-li se v druhém kvadrantu, tak spíná tranzistor T_{HB} nebo T_{DA} . Napětí je kladné a proud záporný. Pokud probíhá řízení zapnutím a vypnutím tranzistoru T_{DA} , pak proud ještě prochází přes diody D_{HA} a D_{DB} . Měnič pracující v třetím kvadrantu, má napětí záporné a proud také, zde pracují tranzistory T_{HB} a T_{DA} . V posledním čtvrtém kvadrantu je napětí záporné a kladný proud. Zde spíná tranzistor T_{HA} nebo T_{DB} . Pokud spínání probíhá tranzistorem T_{DB} , tak proud prochází navíc přes diody D_{HB} a D_{DA} . Jednotlivé průběhy proudu jsou zobrazeny na následujících obrázcích.



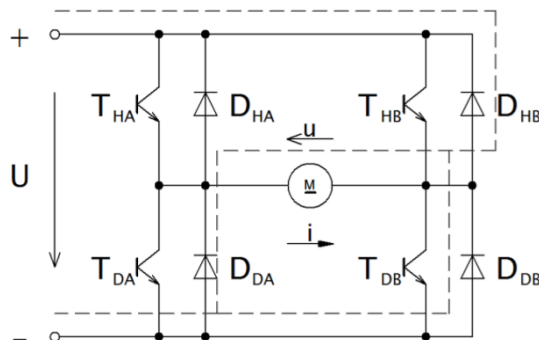
a) Průběh proudu v I. kvadrantu



b) Průběh proudu v II. kvadrantu



c) Průběh proudu v III. kvadrantu



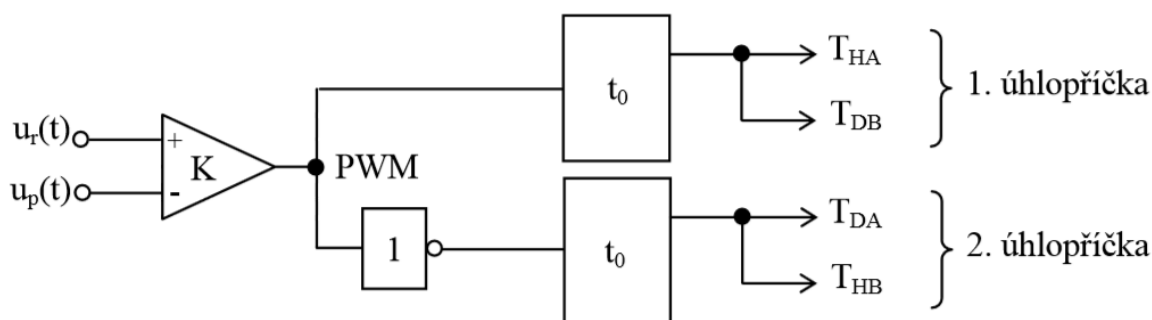
d) Průběh proudu v IV. kvadrantu

Obr. 2.7 Průběh proudu v jednotlivých kvadrantech

Čtyřkvadrantový pulzní měnič je možné řídit unipolárně nebo bipolárně. Rozdíl je pouze v algoritmu (způsobu) zapnutí a vypnutí jednotlivých tranzistorů. Pojmy bipolární a unipolární jsou spjaté s tím, jaké napětí má motor na svých svorkách.

2.4 Bipolární metoda

U bipolární metody je algoritmus řízení následovný: nejdříve sepne dvojice tranzistoru T_{HA} a T_{DB} , to je první úhlopříčka, přičemž druhá dvojice do úhlopříčky je vypnutá, a poté sepne druhá dvojice tranzistorů T_{HB} a T_{DA} a první dvojice je vypnutá. Poté se celý algoritmus opakuje. Pokud by došlo k sepnutí dvou tranzistorů v jedné větvi, to znamená T_{HA} a T_{DA} nebo T_{HB} a T_{DB} , tak by došlo ke zkratu a následnému zničení tranzistorů. Principiální řešení řídicího systému pro bipolární metodu lze vidět na následujícím obrázku.



Obr. 2.8 Princip řídicího systému čtyřkvadrantového PM – bipolární metoda

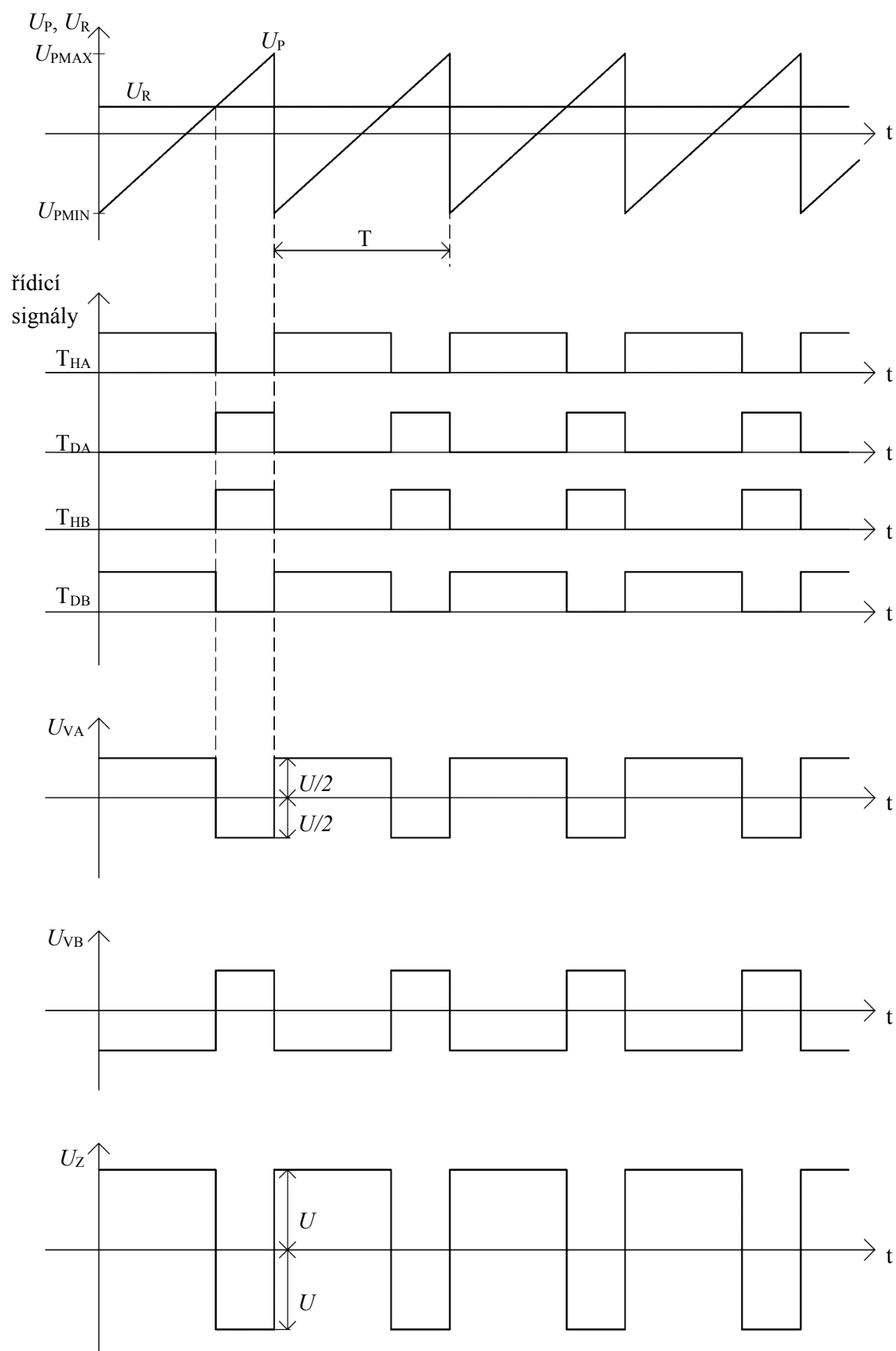
Z obrázku je patrné, že jsou dva vstupní signály. Pilový signál a řídicí signál, kterým se nastavuje požadovaná střída. Tyto dva signály se komparují a za komparátorem je pulzně šířková modulace – PWM signál. Ten se potom posílá do první úhlopříčky. Pro druhou úhlopříčku se tento signál neguje, tím se zajistí, aby nedošlo k sepnutí dvou tranzistorů v jedné větvi. Na konci ještě je blok s názvem t_0 . Ten zajišťuje takzvanou ochrannou dobu, o které byla zmínka už v kapitole 2.1. Aby při přechodu ze zapnutého stavu do vypnutého a naopak nedošlo ke zkratu v jedné větvi. Vypnutí probíhá okamžitě, ale zapnutí je zpožděné o čas t_0 . Tato doba se volí větší, než je vypínací tranzistorová doba.

Bipolární metoda má výhodu v tom, že proud je vždy spojitý, jinak řečeno nemůže dojít k režimu přerušovaných proudů. To proto, že proud má možnost procházet plynule nulou. Nevýhoda této metody je velikost efektivní hodnoty proudu.

Průběhy napětí v celém měniči s bipolárním řízením lze vidět na obrázku 2.9. Z průběhů je patrné, že napětí na zátěži se vypočítá jako rozdíl větvových napětí U_{VA} a U_{VB} .

$$u_Z(t) = u_{VA}(t) - u_{VB}(t)$$

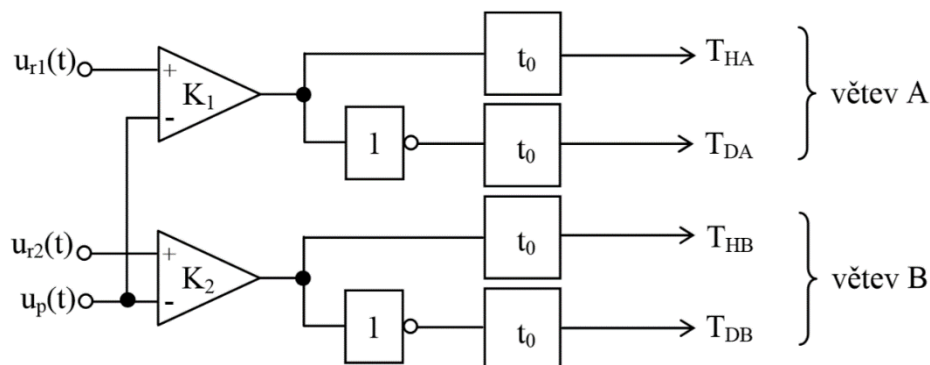
Na zátěži je možné vidět vstupní napětí U_1 a má buď kladnou, nebo zápornou polaritu, proto se tato metoda nazývá bipolární.



Obr. 2.9 Průběhy napětí v celém PM – bipolární metoda

2.5 Unipolární metoda

Druhá metoda je unipolární a oproti bipolární už se nespínají pouze střídavě úhlopříčky, ale jsou fázově posunuté. Algoritmus řízení je následovný: nejdříve sepnou tranzistory T_{HA} a T_{HB} přičemž zbývající dva tranzistory jsou vypnuty, poté dojde k rozepnutí tranzistoru T_{HB} a zároveň k sepnutí tranzistoru T_{DB} . Následovně rozepne tranzistor T_{HA} a sepne T_{DA} . Celý algoritmus se poté opakuje od začátku. Principiální řešení je opět zobrazeno na obrázku.



Obr. 2.10 Princip řídicího systému čtyřkvadrantového PM – unipolární metoda

Jak lze vidět, oproti bipolární metodě zde přibyl jeden řídicí signál navíc, který je symetrický podle vodorovné osy. Pro řídicí signály potom platí vztah:

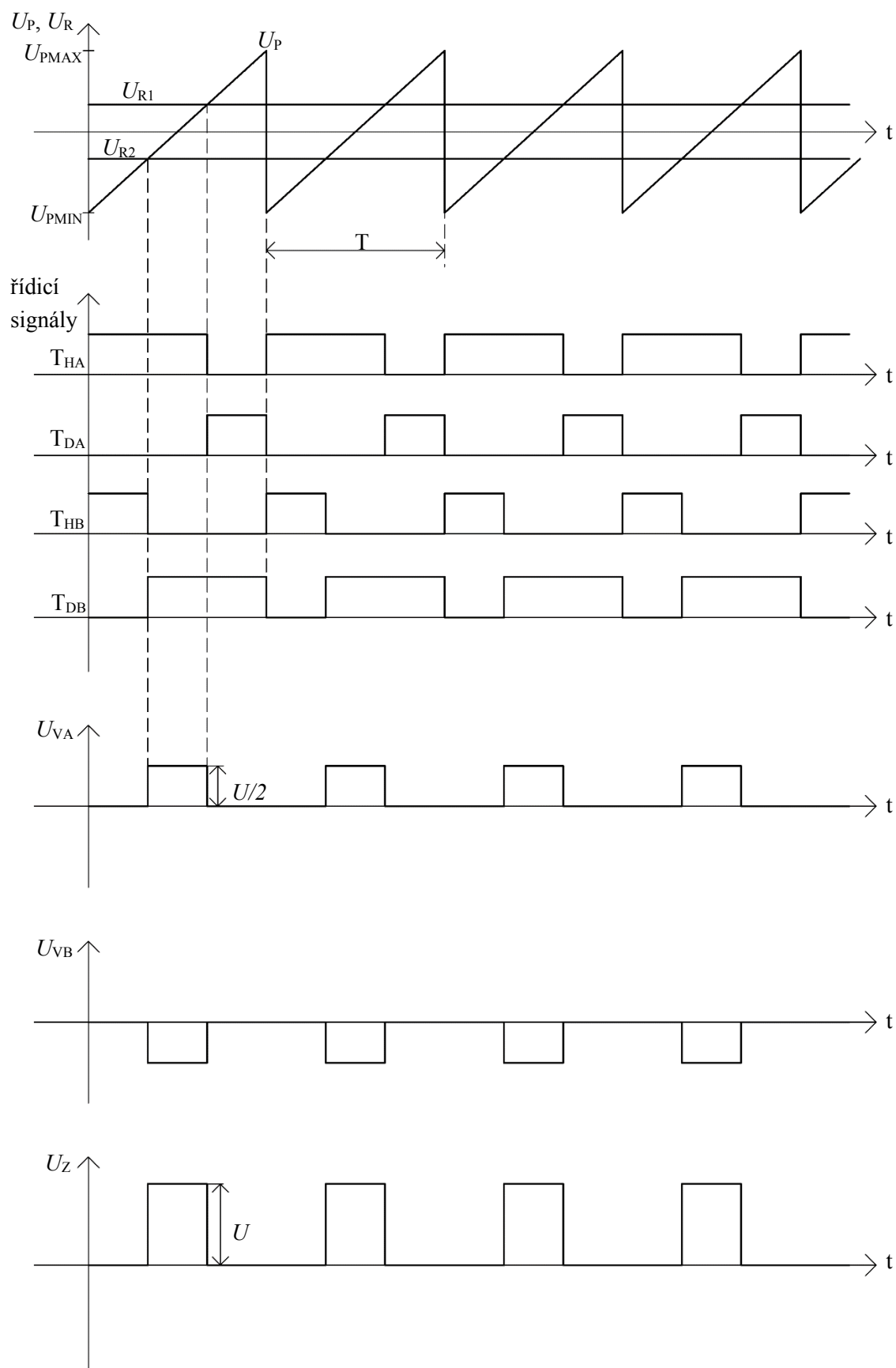
$$u_{r1}(t) = -u_{r2}(t)$$

Poté se oba řídicí signály komparují pilovým signálem. Výstupem je opět pulzně šířková modulace – PWM signál. I zde dochází k negaci jednotlivých signálů. Následně se již všechny čtyři signály pošlou přes blok t_0 do jednotlivých větví. Blok t_0 zde zajišťuje ochrannou dobu, stejně jako u bipolární metody.

Průběhy napětí v celém měniči s unipolárním řízením je i zde možné vidět na obrázku 2.11. Napětí na zátěži se vypočítá stejně jako u bipolární metody, to znamená jako rozdíl větrových napětí U_{VA} a U_{VB} .

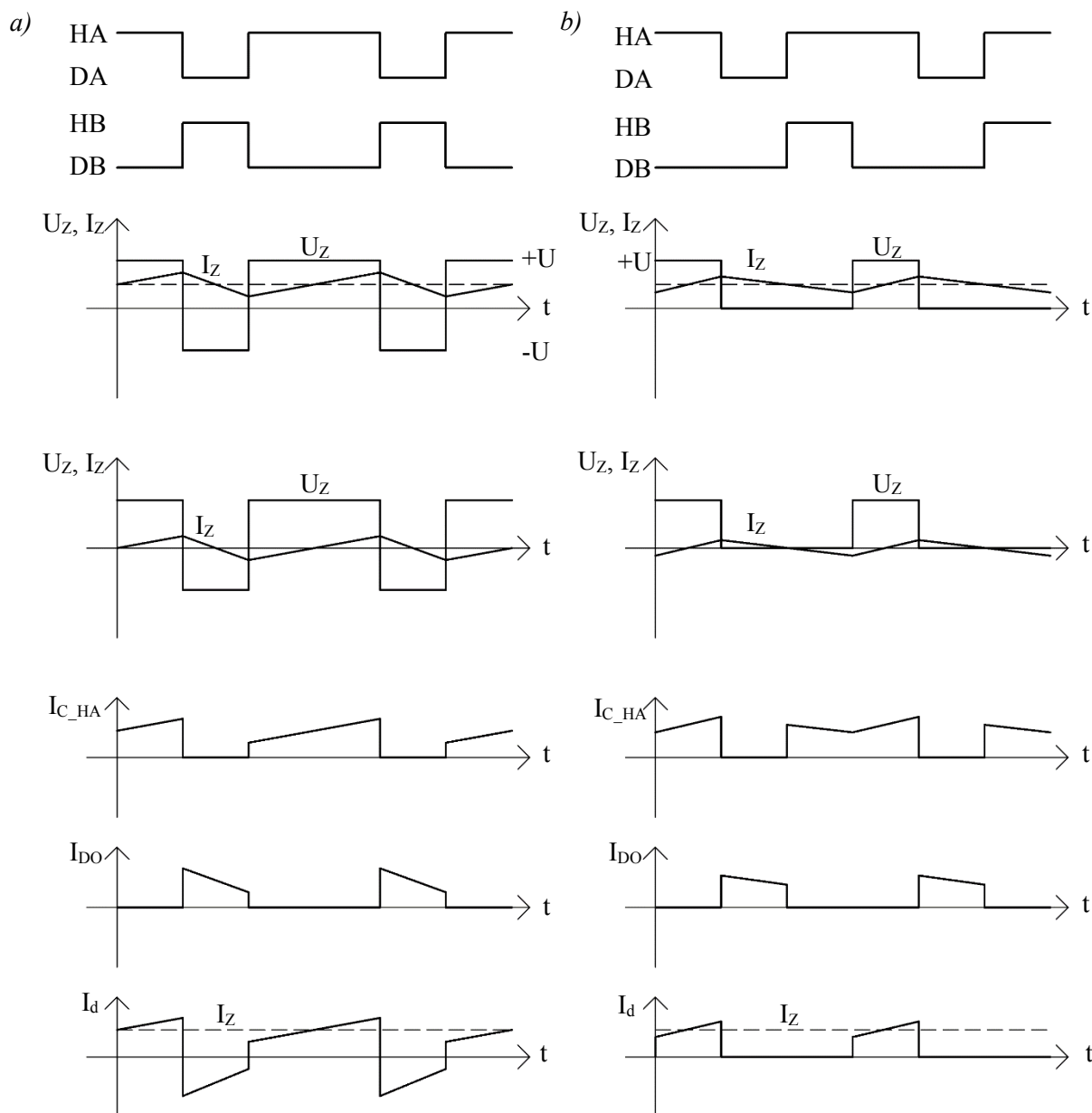
$$u_z(t) = u_{VA}(t) - u_{VB}(t)$$

Tentokrát se na zátěži objeví pouze kladné vstupní napětí U_1 při sepnutí tranzistorů T_{HA} a T_{DB} nebo nulové napětí při sepnutí tranzistorů T_{DA} , T_{DB} , popřípadě T_{HA} , T_{HB} . To platí pro pracovní režim v I. a II. kvadrantu. Pro III. a IV. kvadrant se na zátěži objeví záporné vstupní napětí $-U_1$ při sepnutí tranzistorů T_{HB} a T_{DA} nebo nulové napětí.



Obr. 2.11 Průběhy napětí v celém PM – unipolární metoda

2.6 Porovnání průběhů jednotlivých metod



Obr. 2.12 Průběhy v PM a) Bipolární metoda b) Unipolární metoda

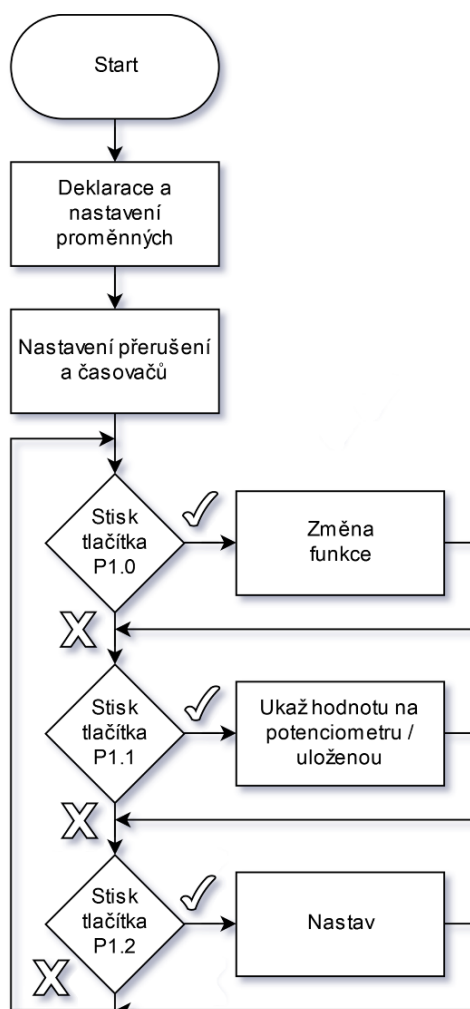
Veškeré průběhy při bipolárním i unipolárním řízení jsou zobrazeny zde na obrázku 2.12. V první části lze vidět řídicí signály, které jsou posílány do tranzistorů. Poté je zobrazeno napětí a proud na zátěži pro spojitý proud a hned pod ním pro nespojitý proud. Střední hodnota napětí a proudu je u obou metod stejná. Efektivní hodnota napětí u bipolární metody nezávisí na střídě, ale u unipolární závisí a to tak, že u unipolární je vždy menší, což má za následek menší vířivé ztráty. U nespojitých proudů je možné vidět, že proud může plynule procházet nulou. Další průběh zobrazuje proud na kolektoru tranzistoru T_{HA} . Střední hodnota je opět stejná, ale špička proudu je u unipolární metody menší. Následný průběh je proud nulové diody. Poslední průběh zobrazuje proud mezilehlého obvodu, to je proud ze zdroje do měniče. Střední hodnota je zase stejná. Zatímco efektivní hodnota proudu u bipolární metody nezávisí na střídě, ale u unipolární ano. A je menší než u bipolární metody.

3. Software

Celý software je napsán v programu Keil a to v programovacím jazyku assembler. Assembler je program, který se používá pro překlad z jazyku symbolických adres, někdy také symbolických instrukcí, do strojového kódu. Program se v jazyku symbolických adres píše po řádcích, stejně jako v jiných jazycích. Jednotlivý řádek má přesně určený formát a to z:

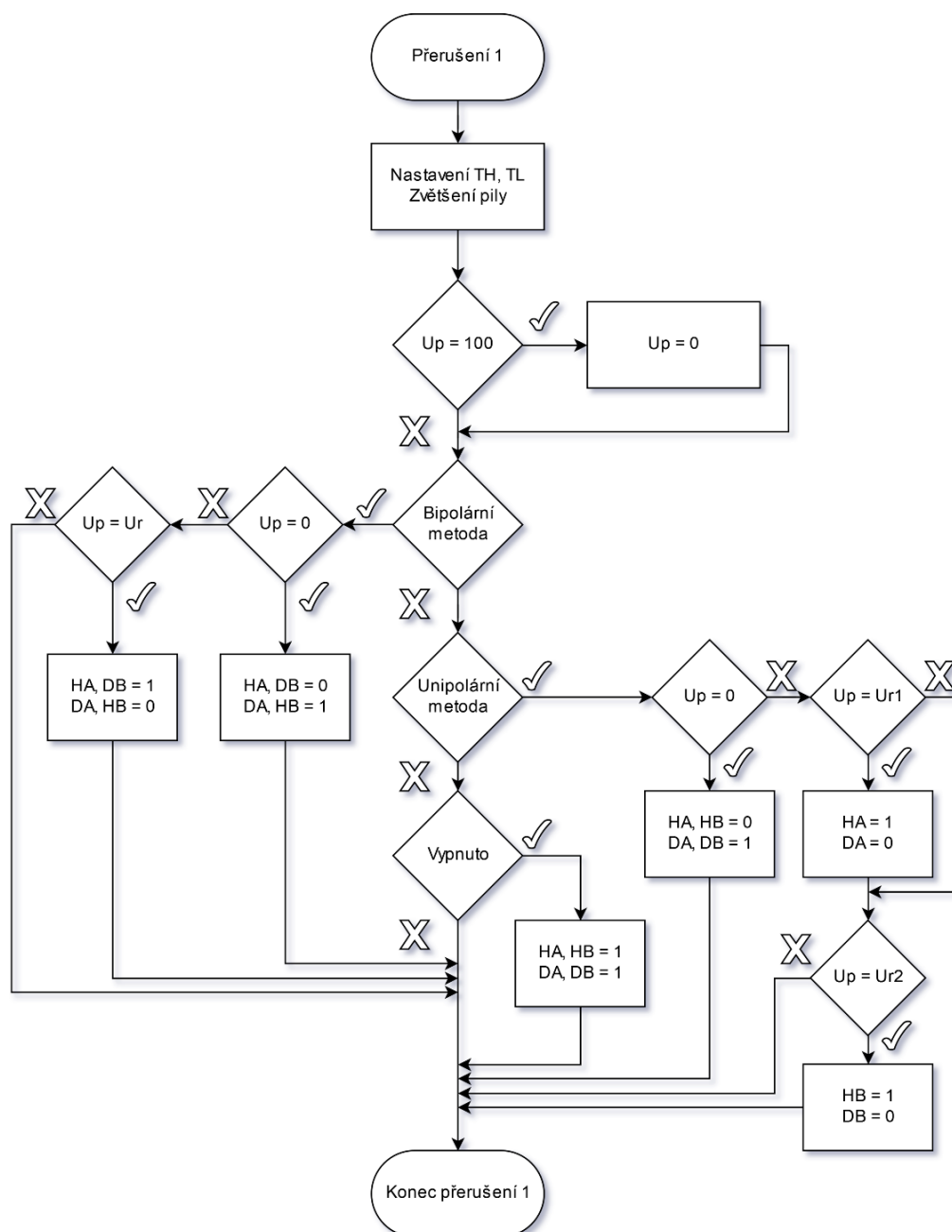
- návěští: posloupnost znaků obsahující písmena od A-Z, číslice 0-9 a podtržítka. Návěští nesmí začínat číslicemi a končí dvojtečkou.
- instrukce: informace, co má vykonat mikroprocesor
- parametr: většina instrukcí požaduje ještě další informaci, nebo informace s kterými pracuje
- poznámka: píše se za středník a může obsahovat cokoliv, překladač tyto poznámky ignoruje.

3.1 Diagram programu



Obr. 3.1 Blokové schéma hlavní smyčky programu

Podle blokového schéma je možné vidět, co se děje v hlavní smyčce. Po zapnutí nebo resetu mikroprocesoru se deklarují veškeré proměnné. Po deklaraci se jednotlivým proměnným přiřadí defaultní hodnota. Následně se nastaví parametry přerušení a časovačů. To znamená, že se určí mód jednotlivých časovačů, povolí se globální přerušení a přerušení od časovače 0 a 1. Nastaví se také hodnoty registrů od jednotlivých čítačů. Až je vše nastaveno, jak má být, tak se spustí časovače a přejde se do hlavní smyčky. Hlavní smyčka je velmi jednoduchá, kontroluje se zde, jestli bylo stisknuté tlačítko, a pokud ano, tak se vykoná příslušná operace. Poté se opět vrátí do smyčky, kde znovu kontroluje tlačítka. Z diagramu je možné vidět, že první tlačítko mění funkce, druhé mění hodnotu na displeji a třetí tlačítko slouží jako nastavovací. Ovládání programu bude popsáno v další podkapitole, a proto operace při stisku tlačítek jsou zde popsány jen stručně.



Obr. 3.2 Blokové schéma 1. přerušení

Jedno přerušení je celé využito pro algoritmus řídicích signálů. Při skoku do přerušení se nejprve nastaví zpět hodnoty registrů TH a TL podle nastavené frekvence z tabulky. Poté se přičte jednička k proměnné Up a zkontroluje se, zda není proměnná rovna hodnotě 100. Pokud ano, dojde k vynulování proměnné. Následuje několik podmínek, které kontrolují, zda se signály mají vytvářet pro bipolární nebo unipolární řízení, popřípadě má-li se vůbec něco generovat. Program je nastaven tak, aby se po restartu negenerovaly impulsy. Také je možné v průběhu vypnout generování impulsů. V případě, že není splněna ani jedna podmínka, skočí se na konec přerušení, kde se program vrátí zpět do hlavní smyčky. K tomu by však nikdy nemělo dojít, protože vždy by měla být právě jedna z podmínek splněna.

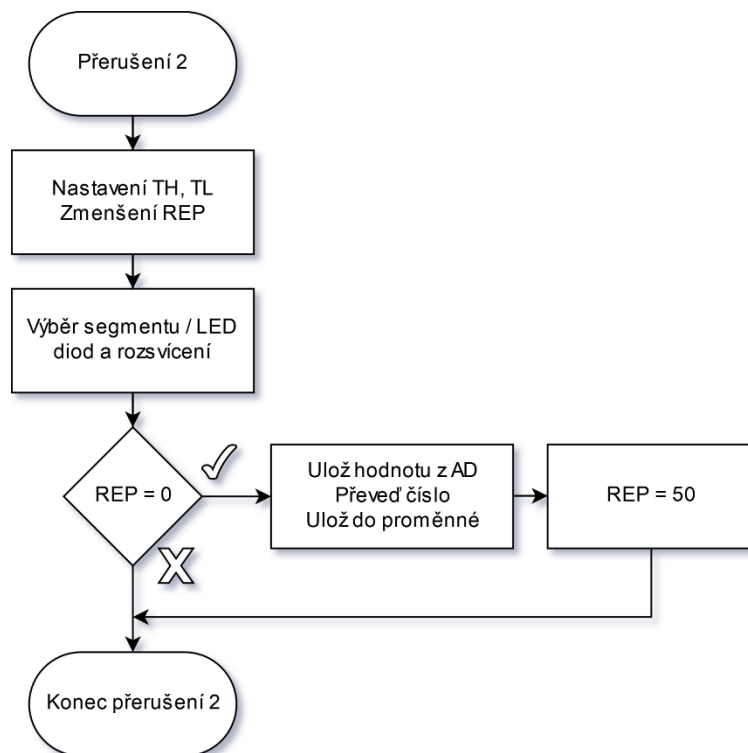
Pokud platí, že je zvolena bipolární metoda, zkontroluje se, zdali je proměnná Up rovna nule. Jestliže ano, tak se pošle na piny P3.2 až P3.5 hodnota 0110. Pokud podmínka splněna není, zkontroluje se další a na piny P3.2 až P3.5 se pošle hodnota 1001, v případě že Up je rovno Ur. Pokud však ani tato podmínka není splněna, nic se nezmění, to znamená, že na portu zůstane stejná hodnota, která byla nastavena posledně a přerušení se ukončí. Toto řízení přesně odpovídá algoritmu, který byl popsán v kapitole bipolární metoda.

Není-li zvolena bipolární metoda, kontroluje se unipolární metoda. Za podmínky, že je Up rovno nule, se na výstup pošle hodnota 0011 a dojde k ukončení přerušení. V případě že není Up rovno nule, ale Up se rovná Ur1, změní se pouze jedna dvojice bitů a to P3.2 na 1 a P3.3 na 0. Následovně dojde k porovnání Up s Ur2 nezávisle na tom, zda byla předchozí podmínka splněna nebo ne. Jestliže i zde dojde k rovnosti, nastaví se druhá dvojice bitů P3.4 na 1 a P3.5 na 0. I zde je postup identický teorii unipolárního řízení.

Poslední možností, kdy není zvolena bipolární metoda, ani unipolární, je vypnuto generování impulsů. V tomto okamžiku se na výstup P3.2 až P3.5 posílají samé jedničky.

Je nutné si uvědomit, že výkonová jednotka má na vstupu negaci. A proto hodnoty, které se posílají z mikroprocesoru, jsou přesně opačné než ty, které je možné vidět v teorii. Proto pro zapnutí posíláme log. 0 a pro vypnutí log. 1.

Hodnoty se posílají na piny P3.2 až P3.5 zcela úmyslně, jelikož na výukovém systému jsou vyvedeny piny P3.2 až P3.5 pro vnější přerušení a pro vnější čítání impulsů pro čítač. V tomto programu však ani jedno z toho není zapotřebí a je možné je využít jako výstup pro řídicí signály.

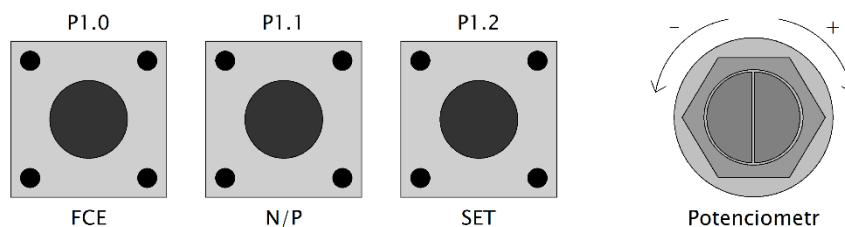


Obr. 3.3 Blokové schéma 2. přerušení

Druhé přerušení, které je zobrazené na obr. 3.3 slouží pro rozsvícení displeje, LED diod a uložení hodnoty z A/D převodníku. V přerušení se nejdříve nastaví hodnoty registrů TH a TL, tentokrát je však pevně stanovený interval a tím je 1ms. Tento čas je dostatečný k rozsvícení všech čtyř pozic displeje a LED diod, aniž by lidské oko poznalo, že v daný okamžik svítí pouze jedna pozice nebo LED diody. Po nastavení registrů se vybere, jestli má svítit displej nebo diody, protože na bráně P2 jsou zapojeny obě periferie. V případě displeje je zapotřebí vzít jedno číslo z dané proměnné, které se má zobrazit a poté jej takzvaně zamaskovat. To proto, že 4 bity určují číslo, které se má zobrazit a další 4 bity určují na které pozici. Jakmile je vše hotové, tak stačí tuto hodnotu poslat na bránu P2 a číslo se rozsvítí. Jelikož jsou LED diody na stejném portu jako displej, je nutné kromě střídání pozic na displeji, ještě mezi tím rozsvěcovat diody. Jelikož by diody svítily pouze 1ms a 4ms displej, vypadalo by to, že diody svítí slabě. Proto se nejdříve rozsvítí první dvě pozice displeje, poté diody, následně zbývající dvě pozice a zase diody. Poté se zkontroluje podmínka, jestli je proměnná nulová. Pokud není, přerušení se ukončí. Jestliže podmínka splněna je, tak se uloží hodnota z A/D převodníku a převede podle funkce. A/D převodník má rozsah od 0-255, ale u střídání bipolární metody je zapotřebí pouze hodnota od 0-100%. U frekvence je rozsah zase nastaven na 50-150Hz. A právě zde se hodnota z potenciometru, pomocí A/D převodníku a následného zpracování převede na potřebný rozsah. Není však zapotřebí, aby se hodnota převáděla co 1ms a tak je zavedena proměnná, která se vždy po splnění podmínky nastaví zpět na hodnotu 50. Tím se zajistí, že převod bude probíhat pouze jednou za 50ms. Tím pádem mikroprocesor není tak moc zatěžován a také se tím zmenší pravděpodobnost, nepříjemného blikání hodnoty na displeji.

3.2 Ovládání programu

Celý program se ovládá pomocí tří tlačítek a jednoho potenciometru, který je umístěný na výukové jednotce. Informace o stavu nám udávají LED diody a displej.

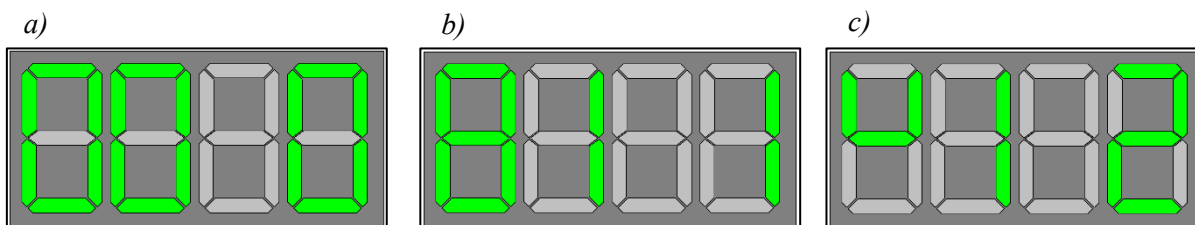


Obr. 3.4 Tlačítka s jejich názvem a potenciometr

Tab. 2 Popis tlačítek

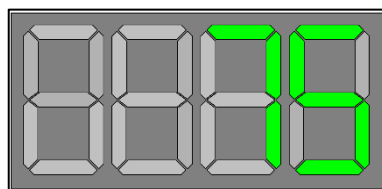
Pin	Zkratka	Význam	Signalizace
P1.0	FCE	Změna funkce – metoda, frekvence, střída	LED č. 7, 6, 5 + displej
P1.1	N/P	Změna zobrazené hodnoty z paměti / potenciometru	LED č. 3, 2 + displej
P1.2	SET	Nastav metodu, frekvenci, střidu	---

Již z diagramu je možné vidět, že první tlačítko slouží ke změně funkce. Tomuto tlačítku jsem dal zkratku FCE. Funkce máme celkem tři a pro každou funkci se rozsvítí příslušná LED dioda. První funkce slouží pro změnu metody a rozsvítí se první dioda s číslem 7. Po restartu je nastavena metoda pro vypnuté generování. Následující metoda je bipolární a poté unipolární. U každé metody se na displeji zobrazí zkratka. Jelikož deska má dekodér, byl jsem použitelnými znaky dost omezen a snažil se o to, aby se aspoň trochu dalo rozpoznat, o jakou metodu jde. U vypnutého generování se zobrazí pouze nuly. Další metoda na displeji zobrazí „BI 1“ jako 1. metoda – bipolární. Pro druhou metodu se vyobrazí „UI 2“, tedy 2. metoda – unipolární. Přepnutí metody se provádí stiskem třetího tlačítka, které nese název SET.



Obr. 3.5 Displej ukazující aktuální metodu a) Vypnuto b) Bipolární c) Unipolární

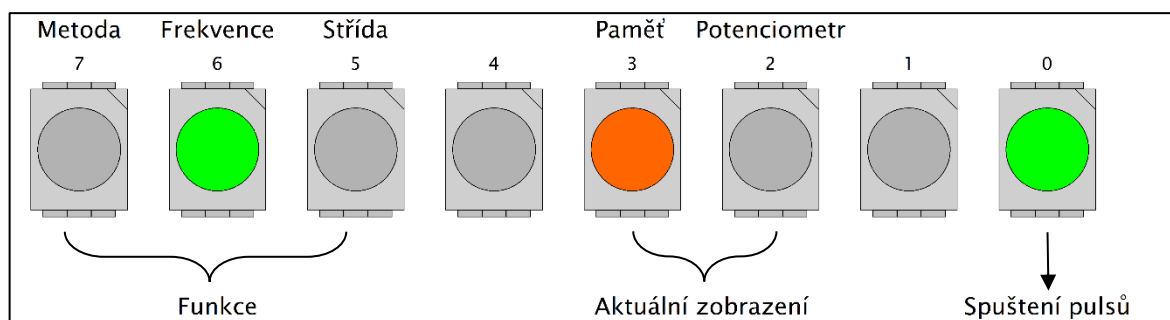
Po zvolení metody se stiskne první tlačítko FCE a může se nastavovat frekvence. Při nastavování frekvence se rozsvítí druhá dioda zleva s číslem 6. Na displeji se zobrazí dvou až třímístné číslo, které udává frekvenci v Hz. Pomocí potenciometru je možné zmenšit nebo zvětšit hodnotu a třetím tlačítkem SET se hodnota nastaví. Zde se může hodit druhé tlačítko, které mění hodnotu na displeji z nastavené na hodnotu z potenciometru. Defaultně je nastaveno, že se na displeji zobrazí hodnota, která je uložena (nastavena) v paměti. Pokud ale bude zapotřebí zobrazit hodnotu, která je zrovna na potenciometru, tak stačí kliknout druhé tlačítko. V případě stisku třetího tlačítka SET by se opět uložila do paměti. K tomu aby se rozlišilo, jestli je na displeji zobrazena uložena hodnota, nebo z potenciometru slouží opět LED diody. A to přesně diody s číslem 3 a 2. Pokud svítí dioda na pozici 3, tak displej ukazuje hodnotu v paměti. Jestliže svítí dioda 2, tak displej ukazuje hodnotu z potenciometru. Po restartu je hodnota frekvence nastavena na 100Hz. Měnit se může v rozmezí od 50Hz do 150Hz s krokem 1Hz. Frekvenci je možné nastavovat, i když je vypnuté generování.



Obr. 3.6 Displej ukazující aktuální frekvenci

Až je nastavena požadovaná frekvence, stiskne se první tlačítko FCE a může se nastavit střída. Podle diody číslo 5 která se rozsvítí, se pozná, že jde o nastavení střídá. Na displeji se opět ukáže hodnota, která odpovídá střídě v %. Měnit ji je možné pomocí potenciometru a uložení probíhá stejně jako v předchozím případě, to znamená třetím tlačítkem SET. I zde je možné přepnout nastavenou hodnotu a hodnotu z potenciometru pomocí druhého tlačítka. Nastavení střídá je možné pouze po zvolení metody bipolární nebo unipolární a to proto, že každá metoda má svou vlastní proměnnou pro střídá. U bipolární metody lze nastavovat hodnotu od 0% do 100% s krokem 1%. Defaultně je nastaveno 50%. Zatímco unipolární metoda dovoluje nastavit střídá od 100% (jeden směr otáčení motoru) do 100% (druhý směr otáčení motoru), ale za to s menším krokem a to 2%. Při restartu je nastavena hodnota na 0%.

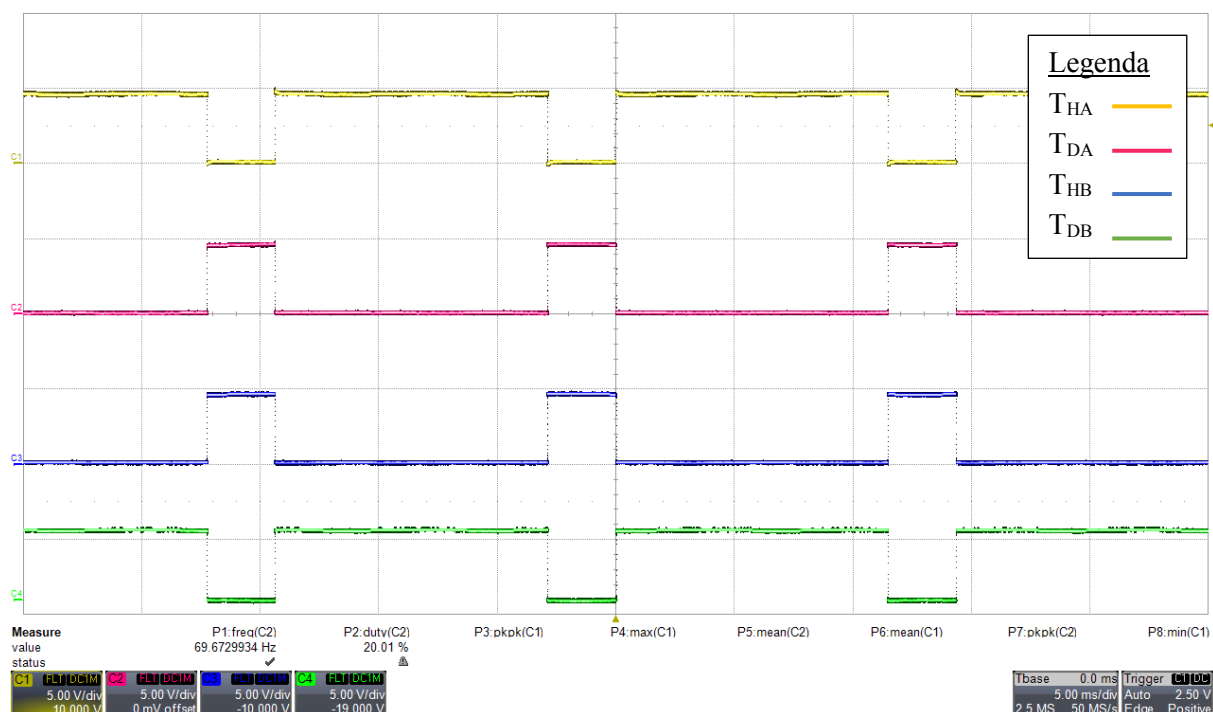
Spuštěné generování impulsů, ať už bipolární nebo unipolární metody, signalizuje poslední dioda s číslem 0.



Obr. 3.7 Signalizace LED diod a jejich význam

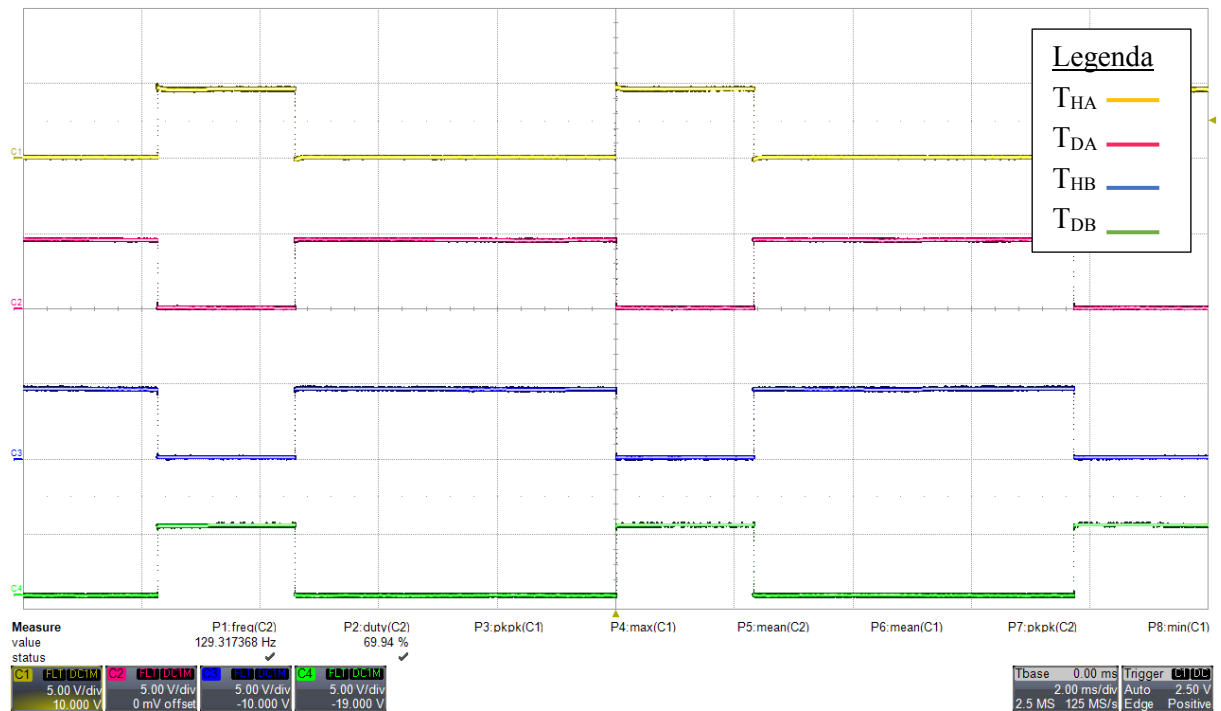
4. Průběhy a ověření

Ověření první části a to funkčnosti softwaru mikroprocesoru probíhalo následovně. Pro napájení výukového systému byl použit kabel USB A-B, pomocí kterého byl rovněž mikroprocesor programován. Na piny P3.2 až P3.5 odkud se poté odváděli řídicí signály do výkonové jednotky, byly připojeny sondy z osciloskopu. Osciloskop byl použit LeCroy 44Mxs-B a to pro veškeré měření. Poté již stačilo nastavit požadované hodnoty a začít měřit. Měřilo se pro bipolární i unipolární řízení. První frekvence byla zvolena 70Hz se střídou 20% a druhá frekvence 130Hz se střídou 70%.

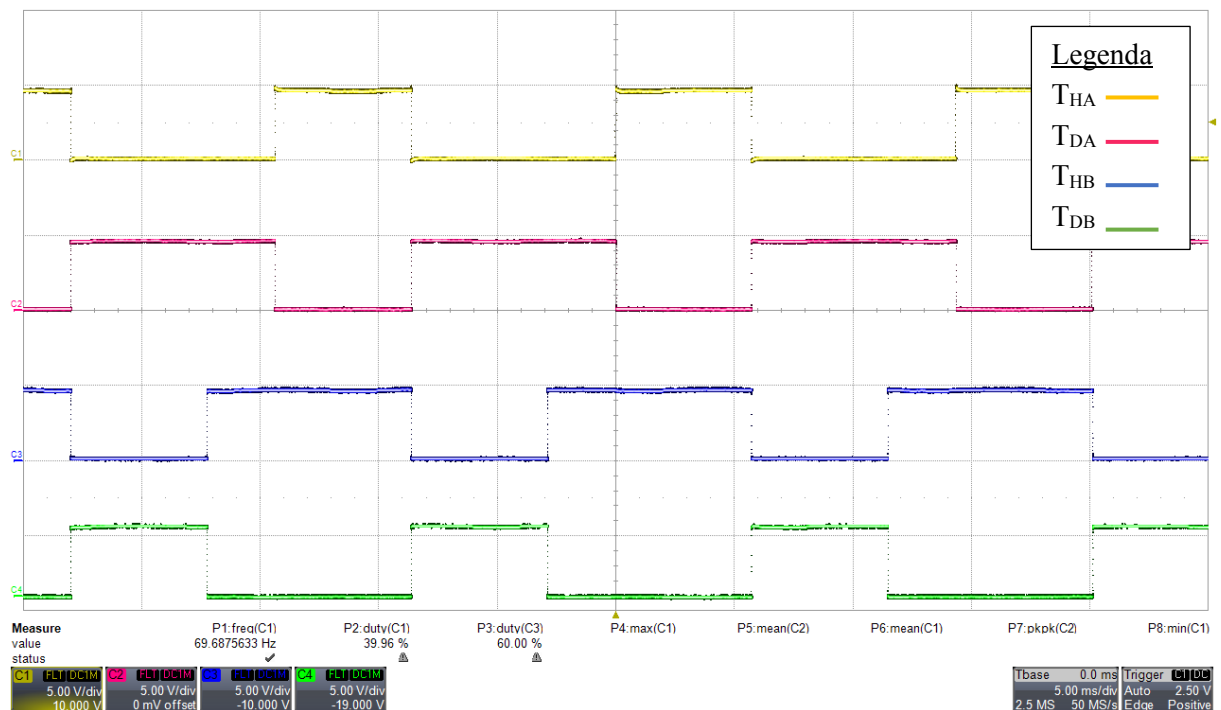


Obr. 4.1 Řídicí signály při bipolárním řízení s frekvencí 70Hz a střídou 20%

4. Průběhy a ověření

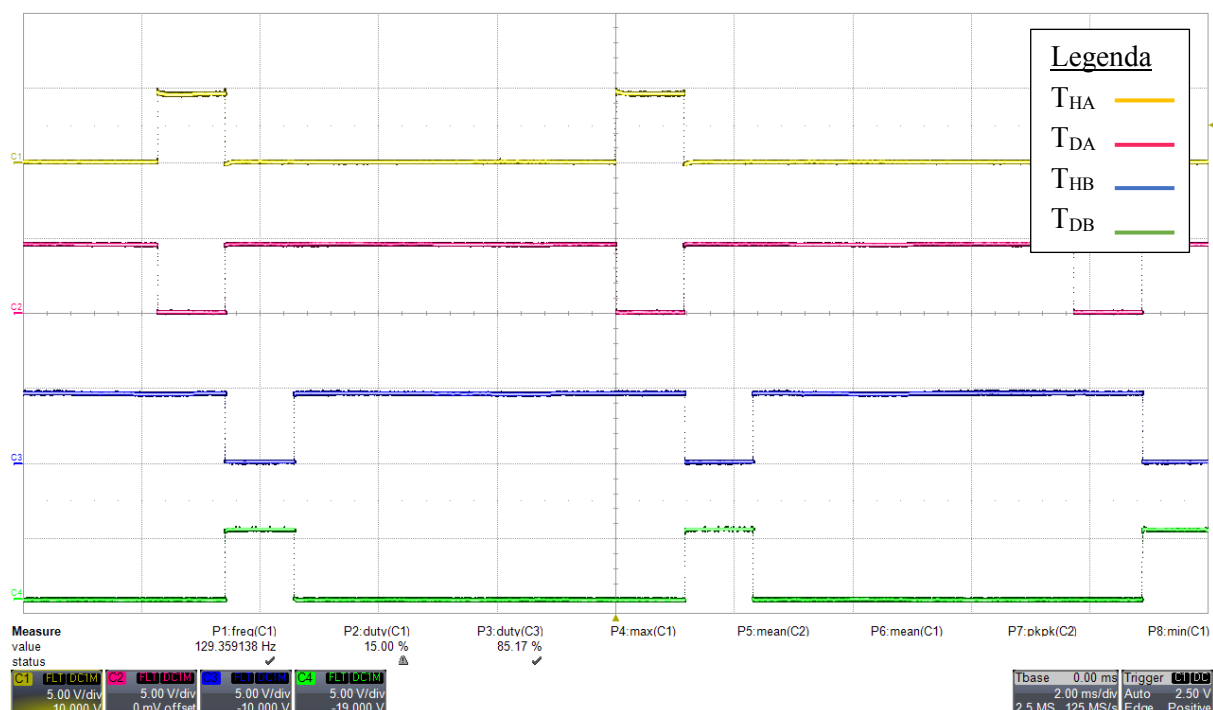


Obr. 4.2 Řídící signály při bipolárním řízení s frekvencí 130Hz a střídou 70%



Obr. 4.3 Řídicí signály při unipolárním řízení s frekvencí 70Hz a střídou 20%

4. Průběhy a ověření



Obr. 4.4 Řídicí signály při unipolárním řízení s frekvencí 130Hz a střidou 70%

Tab. 3 Naměřené hodnoty řídicích signálu – bipolární metoda

Bipolární metoda	Nastaveno	Naměřeno	Odchylka	Odchylka (%)
Frekvence (Hz)	70	69,673	0,327	0,47
	130	129,317	0,683	0,52
Střída (%)	20	20,01	-0,01	0,05
	70	69,94	0,06	0,09

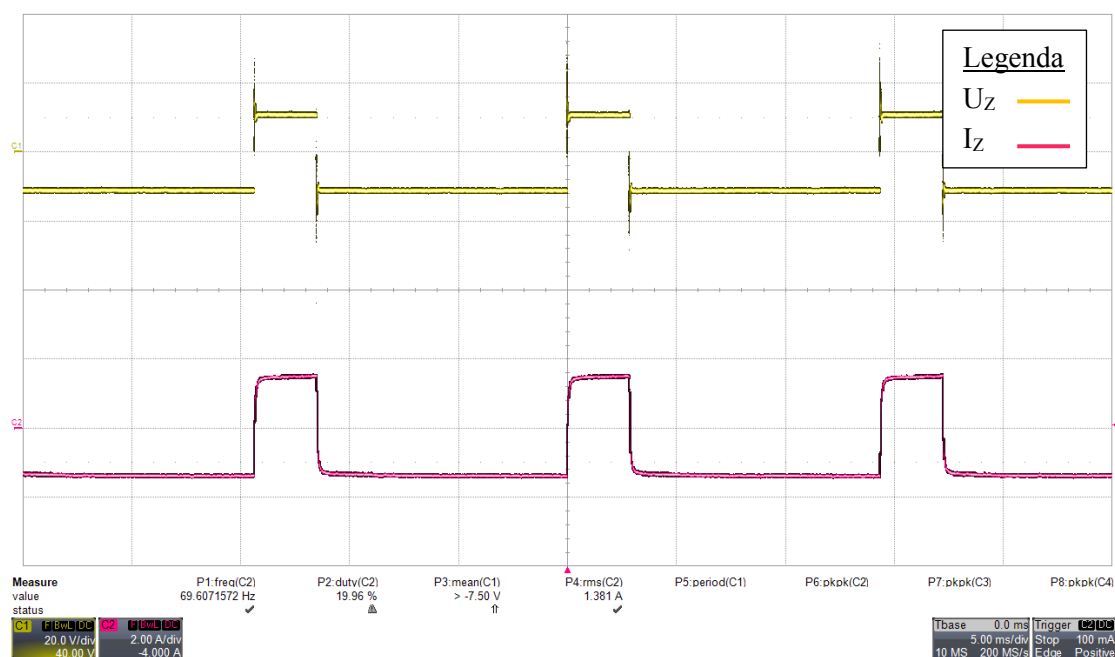
Tab. 4 Naměřené hodnoty řídicích signálu – unipolární metoda

Unipolární metoda	Nastaveno	Naměřeno	Odchylka	Odchylka (%)
Frekvence (Hz)	70	69,688	0,312	0,45
	130	129,359	0,641	0,49
Střída (%)	20	20,04	-0,04	0,2
	70	70,17	-0,17	0,24

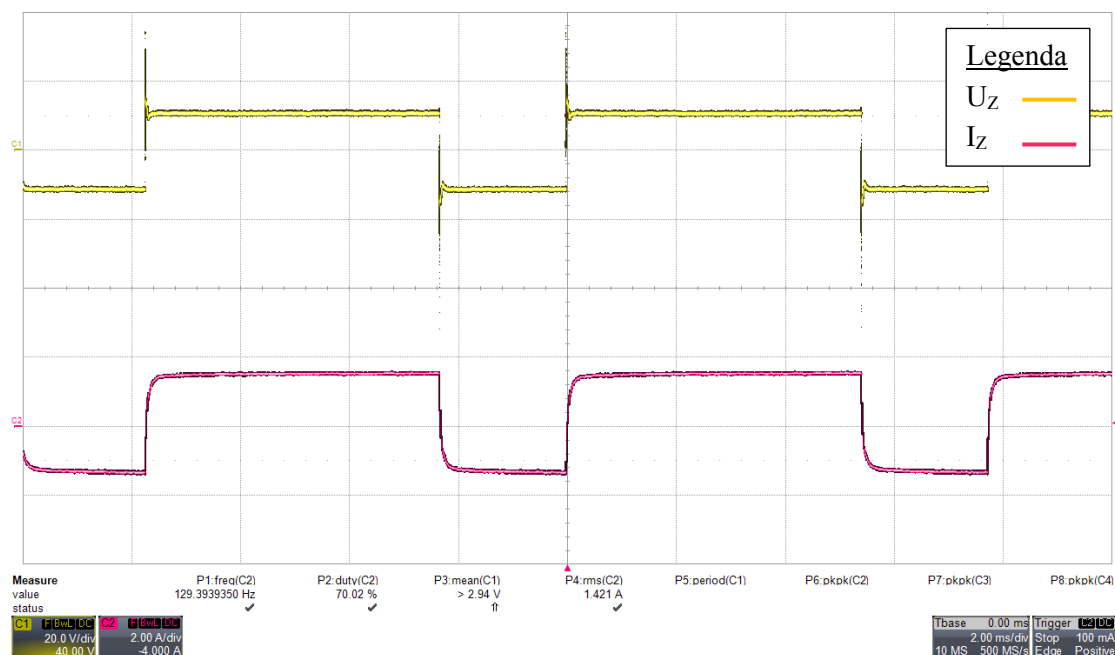
Z průběhů je možné vidět, že řídicí signály odpovídají teoretickým a opravdu na výstupu dostáváme pulzně šířkovou modulaci. Na osciloskopu bylo ještě nastaveno měření frekvence a střidy. Všechny hodnoty jsou sepsány v tabulkách a lze vidět, že naměřené hodnoty jsou téměř shodné s požadovanými. Odchylka je menší než 1%.

4. Průběhy a ověření

Pro druhou část měření se připojila výuková jednotka k výkonové, kde mikroprocesor sloužil jako řídicí jednotka. Výkonová jednotka pro svou funkčnost potřebuje napájecí zdroj s napětím 12V. Na výstup byla připojena odporová zátěž a následně motor. Zátěž byla napájena z autobaterie s napětím 12V. Měření probíhalo opět pro obě metody řízení. Frekvence a střída je také shodná s předchozím měřením. To znamená 70Hz s 20% střídou a 130Hz se střídou 70%. Na osciloskopu bylo nastaveno kromě měření frekvence a střidy, také střední hodnota napětí a efektivní hodnota proudu.

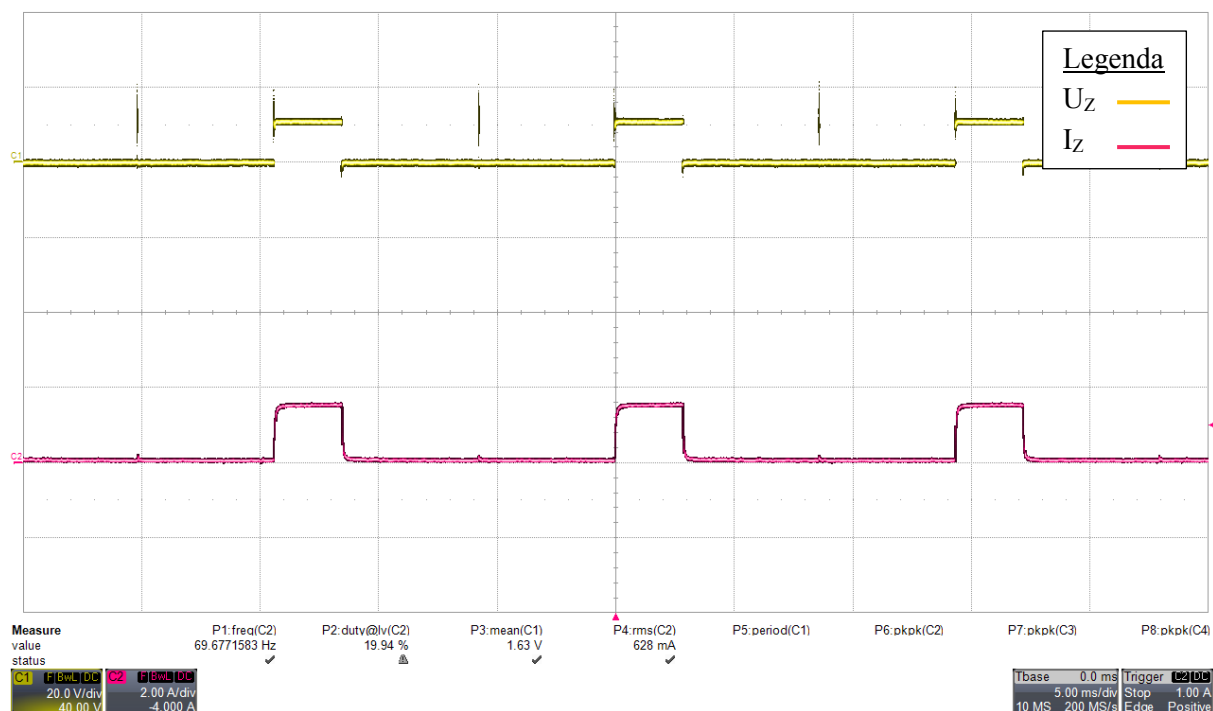


Obr. 4.5 Napětí a proud na odporové zátěži při bipolárním řízení s frekvencí 70Hz a střídou 20%

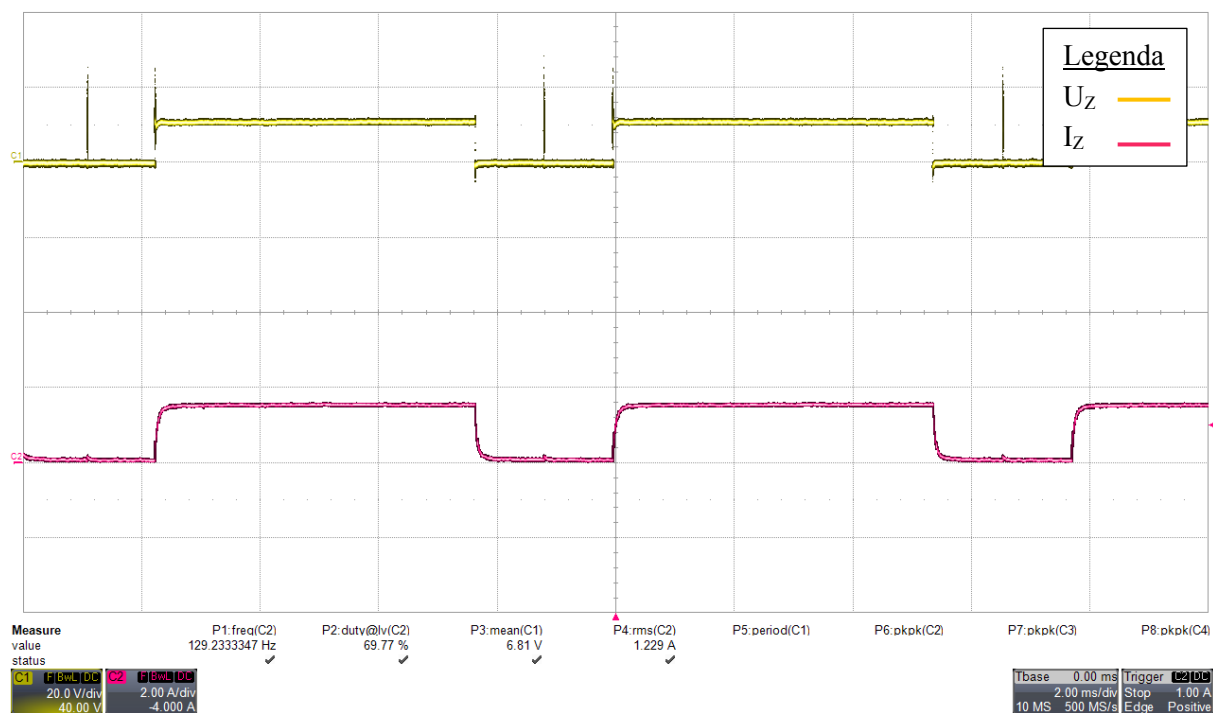


Obr. 4.6 Napětí a proud na odporové zátěži při bipolárním řízení s frekvencí 130Hz a střídou 70%

4. Průběhy a ověření

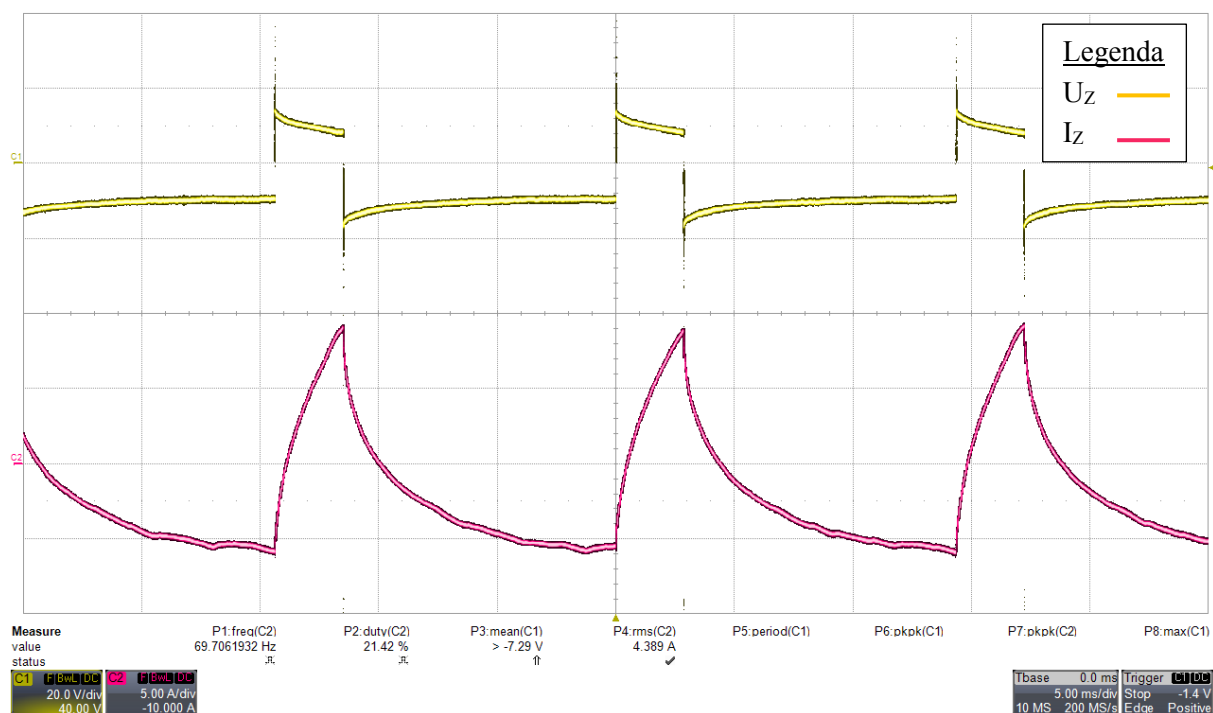


Obr. 4.7 Napětí a proud na odporové zátěži při unipolárním řízení s frekvencí 70Hz a střídou 20%

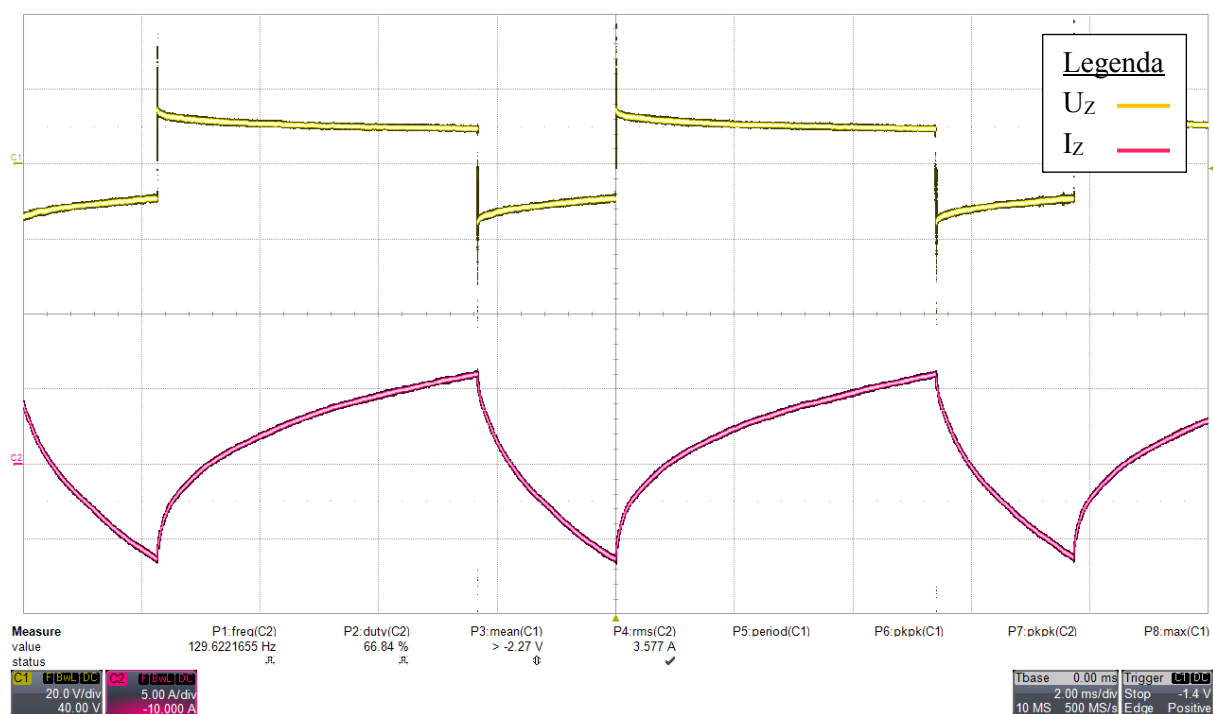


Obr. 4.8 Napětí a proud na odporové zátěži při unipolárním řízení s frekvencí 130Hz a střídou 70%

4. Průběhy a ověření

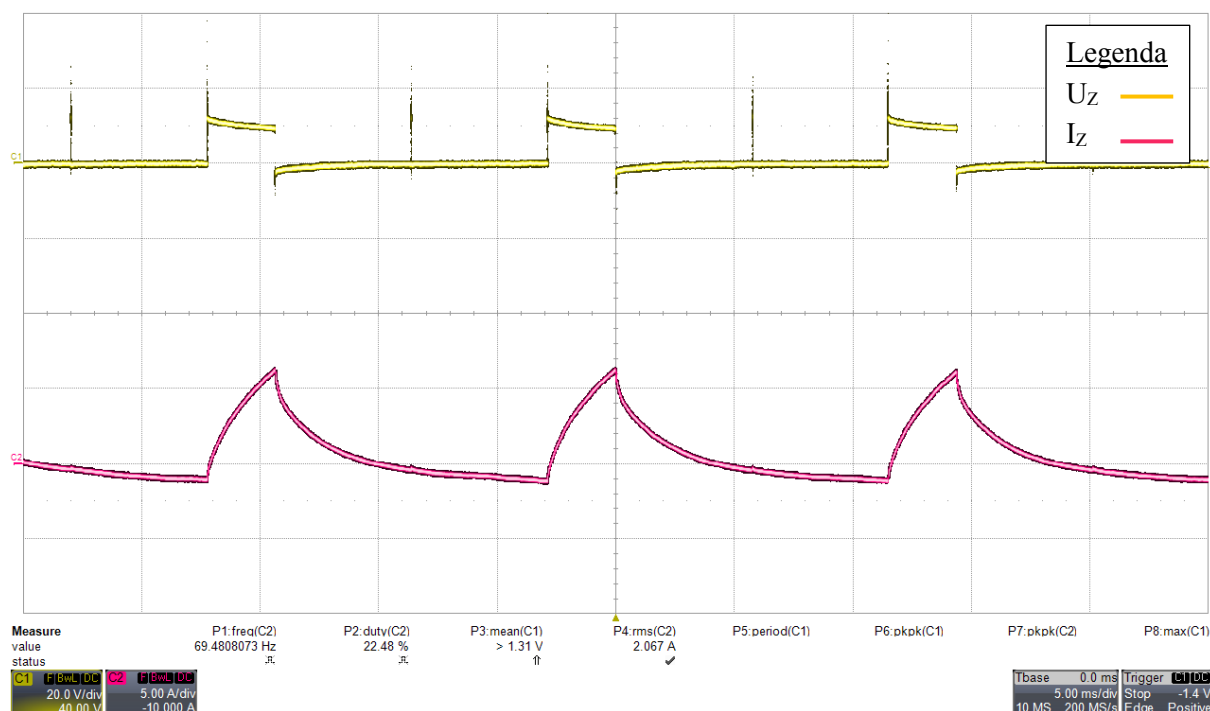


Obr. 4.9 Napětí a proud na motorické zátěži při bipolárním řízení s frekvencí 70Hz a střídou 20%

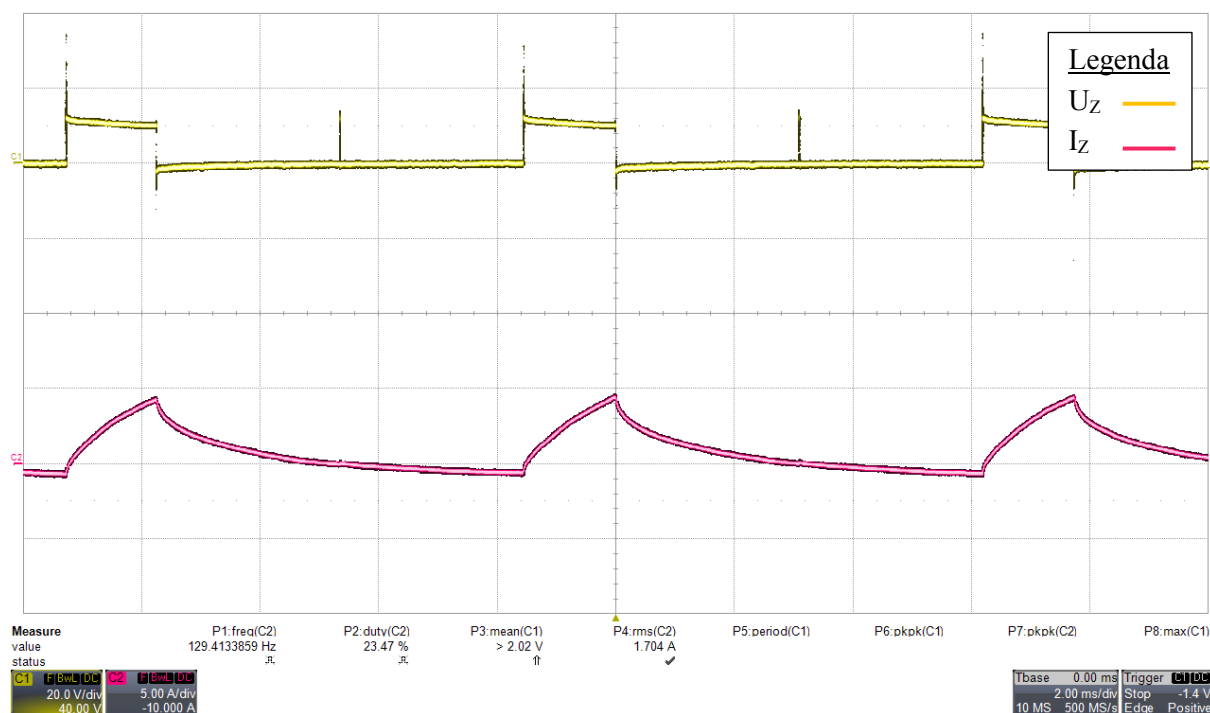


Obr. 4.10 Napětí a proud na motorické zátěži při bipolárním řízení s frekvencí 130Hz a střídou 70%

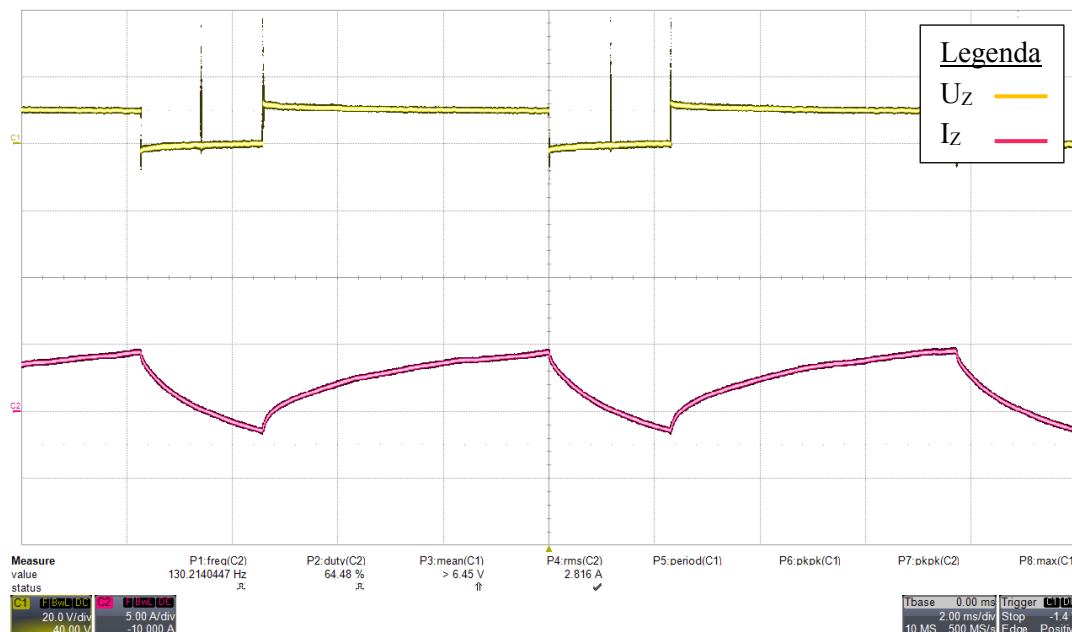
4. Průběhy a ověření



Obr. 4.11 Napětí a proud na motorické zátěži při unipolárním řízení s frekvencí 70Hz a střídou 20%



Obr. 4.12 Napětí a proud na motorické zátěži při unipolárním řízení s frekvencí 130Hz a střídou 20%



Obr. 4.13 Napětí a proud na motorické zátěži při unipolárním řízení s frekvencí 130Hz a střídou 70%

Jak je možné vidět, frekvence opět odpovídá u všech průběhů nastavené frekvenci s odchylkou maximálně 1Hz. Střída taktéž velmi přesně odpovídá nastavené hodnotě, akorát u motorické zátěže nebyl osciloskop nastaven optimálně, proto je možné střidu odečítat pouze přibližně. Pro ověření správného nastavení je to však dostačující. V napětí se objevují špičky vlivem spínání a rozpínání tranzistorů.

Z obrázků je viditelné napětí, které odpovídá teorii. To znamená, že pro bipolární řízení nabývá hodnot +12V a -12V. U unipolárního řízení na výstupu dostáváme pouze +12V a 0V. Střidu je možné nastavit i pro druhý směr otáčení. To znamená, že výstupní napětí bude nabývat hodnot -12V a 0V. Proud u odporové zátěže kopíruje napětí a má taktéž obdélníkový průběh. Zatímco u motorické zátěže, kterou lze považovat za RL obvod s protinapětím, napětí roste a klesá exponenciálně. Z průběhů je také viditelné, že u bipolárního řízení je špičkový proud daleko větší. Příčinou většího zvlnění je to, že bipolární metoda nabývá hodnot +U a -U, ale unipolární pouze +U a 0V nebo -U a 0V. Pro frekvenci 70Hz a střidu 20% dosahuje až 9A. Zatímco u unipolárního řízení je špičková hodnota menší a to 6A. Totožné je i menší zvlnění proudu při unipolárním řízení. Na obrázku 4.11 a 4.12 jsou ještě zobrazeny průběhy pro motorickou zátěž při stejné střídě 20%, ale rozdílné frekvenci. Jak je možné vidět, čím větší frekvence, tím menší zvlnění proudu a také menší hodnota efektivního proudu.

U unipolární metody je možné vidět několik špiček napětí. A to proto, že v každé špičce dochází k sepnutí a rozepnutí tranzistorů. Když sepne tranzistor T_{HA} a T_{DB} , tak je na výstupu kladné napětí. Poté sepnou tranzistory T_{DA} a T_{DB} a na výstupu je nulové napětí. Následně T_{HA} a T_{HB} a výstupní napětí je také nulové. K sepnutí tranzistorů nedochází okamžitě, a proto se na výstupu velmi krátce objeví špičkové napětí. U bipolární metody spínají tranzistory T_{HA} a T_{DB} a poté T_{HB} a T_{DA} , proto se na výstupu neobjeví tři špičky, jako u unipolární metody, ale pouze dvě.

Závěr

Účelem této bakalářské práce bylo vytvořit software pro mikroprocesor řady 8051, který řídil výkonovou jednotku pulzního měniče. Na výstup pulzního měniče se poté připojila zátěž a to konkrétně odporová a motorická. V teoretické části byl popsán mikroprocesor a periferie, kterými disponuje výukový systém. Mou snahou bylo využít co nejvíce a zároveň co nejefektivněji veškeré periferie. Poté byl proveden rozbor pulzního měniče, jak je realizován a jaké možnosti řízení jsou k dispozici. Metody řízení jsou dvě a to bipolární a unipolární. Pro obě metody byl sepsán software, pomocí kterého se řídila výkonová část. Jelikož je software poměrně rozsáhlý a ovládání může působit chaoticky, je zde podrobně popsáno, jak je možné program ovládat. Také je zde diagram, z kterého je patrné, jak program funguje. Poslední částí bylo zaznamenání důležitých veličin. Konkrétně šlo o řídicí signály, které vysílal mikroprocesor a poté výstupní napětí a proud na zátěži. Veškeré průběhy byly zachyceny pomocí osciloskopu. Na obrázcích s řídicími signály je možné vidět, že řídicí jednotka funguje přesně podle teorie a odchylka mezi žádanou a skutečnou frekvencí i střídou je menší než 1%. Průběhy napětí a proudu potom zachycují chování pulzního měniče, pokud se změní metoda, frekvence, anebo střída. Z teorie již plyne, že unipolární metoda oproti bipolární a vyšší frekvence má vliv na zmenšení zvlnění proudu. Střídou se pak mění střední hodnota napětí, čímž se mění otáčky i směr motoru. Výukový systém se softwarem bude sloužit pro demonstrační účely v laboratorních cvičeních.

Tato bakalářská práce byla velmi přínosná pro pochopení funkčnosti pulzního měniče a jeho chování při různých metodách řízení. Taktéž z pohledu na mikroprocesor a dokázání čeho všeho je schopný, takový obyčejný mikrokontrolér AT89S52, který je v dnešní době k dostání již za pár desítek korun. Do budoucna by se mohl využít výkonnější mikroprocesor, který by pracoval s větší frekvencí, a také by mohla být zavedena zpětná vazba, pomocí které by bylo možné regulovat proudovou nebo rychlostní smyčku, popřípadě oboje.

Použitá literatura

1. Miroslav Patočka, *Vybrané stati z výkonové elektroniky*, Svazek II, Brno 1997
2. František Vondrášek, *Výkonová elektronika*, Svazek III, Západočeská univerzita, Plzeň 1998
3. Petr Palacký, *Mikroprocesorové řídicí systémy I*, Ostrava 2004
4. Tomáš Pavelek, Václav Sládeček, *Výkonová elektronika KE – učební texty*, Ostrava 2005
5. Martin Škopek, *Inovace laboratorního stanoviště se stejnosměrným regulovaným pohonem – Diplomová práce*, Ostrava 2006
6. Petr Palacký, *Číslicová a mikroprocesorová technika – učební texty*, Ostrava 2007
7. Datasheet *8-bit Microcontroller with 8K Bytes In-System Programmable Flash - AT89S52*
[online] Dostupné z: <http://www.atmel.com/images/doc1919.pdf>

Seznam příloh

Příloha č. I: Software pro ověření funkčnosti řízení jednotky